



TUGAS AKHIR – TE 141599

## ***MOBILE* ROBOT BERTENAGA SURYA UNTUK PEMETAAN KONSENTRASI GAS**

Yusril Izza  
NRP 07111645000015

Dosen Pembimbing  
Dr. Muhammad Rivai, S.T., MT.  
Fajar Budiman, S.T., M.sc

DEPARTEMEN TEKNIK ELEKTRO  
Fakultas Teknologi Elektro  
Institut Teknologi Sepuluh Nopember  
Surabaya 2018





TUGAS AKHIR – TE 141599

***MOBILE* ROBOT BERTENAGA SURYA UNTUK  
PEMETAAN KONSENTRASI GAS**

Yusril Izza  
NRP 07111645000015

Dosen Pembimbing  
Dr. Muhammad Rivai, S.T., MT.  
Fajar Budiman, S.T., M.sc

DEPARTEMEN TEKNIK ELEKTRO  
Fakultas Teknologi Elektro  
Institut Teknologi Sepuluh Nopember  
Surabaya 2018

.....*Halaman ini sengaja dikosongkan.*.....



*FINAL PROJECT – TE 141599*

***SOLAR POWERED MOBILE ROBOT FOR GAS  
CONCENTRATION MAPPING***

Yusril Izza  
NRP 07111645000015

*Supervisor*  
Dr. Muhammad Rivai, S.T., MT.  
Fajar Budiman, S.T., M.sc

***ELECTRICAL ENGINEERING DEPARTMENT***  
*Faculty of Electrical Technology*  
*Sepuluh Nopember Institute of Technology*  
*Surabaya 2018*

.....*Halaman ini sengaja dikosongkan*.....

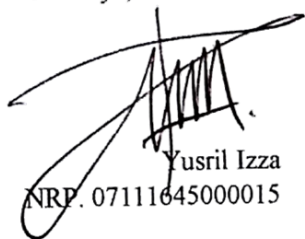
## **PERNYATAAN KEASLIAN TUGAS AKHIR**

Dengan ini saya menyatakan bahwa isi sebagian maupun keseluruhan Tugas Akhir saya dengan judul "*MOBILE ROBOT BERTENAGA SURYA UNTUK PEMETAAN KONSENTRASI GAS*" adalah benar-benar hasil karya intelektual sendiri, diselesaikan tanpa menggunakan bahan-bahan yang tidak diijinkan dan bukan merupakan karya orang lain yang saya akui sebagai karya sendiri.

Semua referensi yang dikutip maupun dirujuk telah ditulis secara lengkap pada daftar pustaka.

Apabila ternyata pernyataan ini tidak benar, saya bersedia menerima sanksi sesuai dengan peraturan yang berlaku.

Surabaya, 26 Juni 2018



Yusril Izza  
NRP. 07111645000015

.....*Halaman ini sengaja dikosongkan*.....



**MOBILE ROBOT BERTENAGA SURYA UNTUK  
PEMETAAN KONSENTRASI GAS**

**TUGAS AKHIR**

**Diajukan Untuk Memenuhi Sebagian Persyaratan  
Untuk Memperoleh Gelar Sarjana Teknik  
Pada**

**Bidang Studi Elektronika  
Departemen Teknik Elektro  
Fakultas Teknologi Elektro  
Institut Teknologi Sepuluh Nopember**

**Menyetujui :**

**Dosen Pembimbing I**

**Dosen Pembimbing II**

**Dr. Muhammad Rival, S.T., M.T.      Pajar Budiman, S.T., M.Sc**  
**NIP. 1969 04 26 1994 03 1003      NIP. 1986 07 07 2014 04 1001**



.....*Halaman ini sengaja dikosongkan*.....

## ***Mobile Robot Bertenaga Surya Untuk Pemetaan Konsentrasi Gas***

Nama : Yusril Izza  
Pembimbing : 1. Dr. Muhmmad Rivai, S.T., M.T.  
2. Fajar Budiaman, S.T., M.Sc

### **ABSTRAK**

Pemetaan kadar konsentrasi gas pada eksploitasi pertambangan minyak sangat berbahaya dilakukan oleh manusia. Diperlukannya suatu alat yang dapat melakukannya tanpa kontaminasi terhadap kadar gas dan mampu berkerja lebih tahan lama yang salah satunya adalah sebuah robot. Pada penelitian ini telah dibuat sebuah *mobile* robot bertenaga surya untuk pemetaan konsentrasi gas yang dapat berjalan secara otomatis dengan sistem *navigation tracking waypoint*. Pada robot ini dilengkapi Sensor Gas TGS 2600, Sensor GPS Ublox Neo-M8N, Sensor *Compass* HMC5883L, Mikrokontroler Arduino Mega2560, *Raspberry* Pi, database MySQL phpMyAdmin, jaringan komunikasi *wireless*, web hosting yang terintegrasi dengan Google Maps dan modul panel surya sehingga bisa berkerja lebih tahan lama. Hasil penelitian menunjukan *Mobile Robot* mampu memetakan dengan keakuratan rata-rata 3 meter, kadar konsentrasi gas lebih dari 20 ppm secara *online realtime* dan mampu menampilkannya dalam bentuk web hosting yang terintegrasi dengan Google Map. Adanya panel surya, sistem mampu berkerja lebih tahan lama 65 menit dibandingkan dengan baterai tanpa panel surya. Dari hasil penelitian tersebut dapat ditunjukan bahwa *Mobile Robot* mampu berkerja lebih tahan lama dalam memetakan konsentrasi gas pada titik-titik yang telah ditentukan, dan hasil pemetaan konsentrasi gas dapat ditampilkan pada website yang dapat diakses melalui internet.

Kata kunci: *Mobile Robot*, Panel Surya, Navigasi, Pemetaan, *waypoint*.

.....*Halaman ini sengaja dikosongkan*.....

## **SOLAR POWERED MOBILE ROBOT FOR GAS CONCENTRATION MAPPING**

Name : Yusril Izza  
Supervisor : 1. Dr. Muhmmad Rivai, S.T., M.T.  
2. Fajar Budiaman, S.T., M.Sc

### **ABSTRACT**

*Mapping the concentration of gas in the exploitation of oil mining is very dangerous to be done by humans. It requires a tool that can do without contamination of gas content and can work more durable one of which is a robot. In this research, a solar powered mobile robot has been developed for mapping gas concentrations that can run automatically with a navigation tracking waypoint system. In this robot equipped with TGS 2600 Sensor Gas, Neo-M8N Ublox GPS Sensor, HMC5883L Compass Sensor, Arduino Mega2560 Microcontroller, Raspberry Pi, MySQL phpMyAdmin database, wireless communication network, web hosting integrated with Google Maps and solar panel module so it can work more durable. The results show Mobile Robot able to map with an average accuracy of 3 meters, gas concentration levels of more than 20 ppm in realtime online and able to display it in the form of web hosting that is integrated with Google Map. The existence of solar panels, the system is able to work more durable 65 minutes compared to batteries without solar panels. From the results of this study can be shown that Mobile Robot able to work more durable in mapping gas concentration at the points that have been determined, and the results of gas concentration mapping can be displayed on websites that can be accessed via the internet.. From the results of this study can be shown that Mobile Robot able to work more durable in mapping gas concentration at the points that have been determined, and the results of gas concentration mapping can be displayed on websites that can be accessed via the internet.*

**Keywords:** *Mobile Robot, Solar Panel, Navigation, Mapping, Waypoint.*

.....*Halaman ini sengaja dikosongkan*.....

## KATA PENGANTAR

Puji syukur alhamdulillah penulis panjatkan ke hadirat Allah SWT yang telah melimpahkan rahmat dan karunia-Nya, sehingga penulis dapat menyelesaikan dan menyusun penelitian Tugas Akhir dengan judul “*MOBILE ROBOT BERTENAGA SURYA UNTUK PEMETAAN KONSENTRASI GAS*” tepat pada waktunya. Shalawat dan salam selalu tercurah kepada Rasulullah Muhammad SAW yang selalu menjadi panutan umat manusia sampai akhir zaman.

Dukungan dan semangat yang diberikan dari berbagai pihak sangat membantu penulis dalam menyelesaikan penelitian Tugas Akhir ini, ucapan terimakasih penulis sampaikan kepada :

1. Dr. Muhammad Rivai, ST., MT. dan Fajar Budiman, S.T., M.Sc. selaku dosen pembimbing, atas dukungan moril maupun materiil, serta bimbingan, inspirasi, pengarahan, dan motivasi yang diberikan selama pengerjaan penelitian Tugas Akhir ini.
2. Dr. Ir. Hendra Kusuma, M. Eng. Sc., Ir. Harris Pirngadi, MT., Dr. Eng. Ir. Totok Mujiono, M.Ikom., Muhammad Attamimi, B.Eng., M.Eng. PhD. selaku dosen penguji atas pengarahan dan masukan yang diberikan selama pengerjaan buku laporan penelitian Tugas Akhir ini.
3. Dr. Eng. Ardyono Priyadi, ST., M.Eng. Selaku Ketua Departemen Teknik Elektro Fakultas Teknologi Elektro.
4. Seluruh dosen-dosen Departemen Teknik Elektro, khususnya dosen-dosen bidang studi Elektronika.
5. Mama dan kakak tercinta yang selalu memberikan dukungan baik berupa moril maupun materiil sehingga laporan proyek akhir ini bisa terwujud.
6. Rekan-rekan yang banyak membantu dalam penyelesaian tugas akhir ini.

Kiranya demikian penulis hanya dapat mendoakan semoga budi baik dari semua pihak tersebut di atas mendapat imbalan dari Allah SWT.

Dalam penyusunan Laporan proyek akhir ini, penulis menyadari bahwa laporan ini jauh dari sempurna, oleh karena itu penulis mengharapkan saran dan kritik dari pembaca demi kesempurnaan laporan nantinya.

Akhir kata penulis berharap semoga Laporan Tugas Akhir ini dapat bermanfaat bagi pembaca, khususnya mahasiswa Teknik Elektronika dan mahasiswa Teknik Elektro pada umumnya.

Surabaya, 21 Juni 2018

Yusril Izza



## DAFTAR ISI

PERNYATAAN KEASLIAN .....	i
TUGAS AKHIR .....	iii
ABSTRAK .....	v
ABSTRACT .....	vii
KATA PENGANTAR .....	ix
DAFTAR ISI .....	xi
DAFTAR GAMBAR .....	xv
DAFTAR TABEL .....	xvii
DAFTAR RUMUS .....	xix
BAB I. PENDAHULUAN .....	1
1.1. Latar Belakang .....	1
1.2. Perumusan Masalah .....	2
1.3. Tujuan Penelitian .....	2
1.4. Batasan Masalah .....	2
1.5. Metodologi Penelitian .....	3
1.6. Sistematika Penulisan .....	4
1.7. Relevansi .....	4
BAB II. DASAR TEORI .....	5
2.1. Sensor Semikonduktor .....	5
2.2. Prinsip Kerja Sensor Semikonduktor .....	5
2.3. Sensor Gas TGS2600 .....	6
2.4. Arduino Mega 2560 .....	8
2.4.1. Spesifikasi Arduino Mega 2560 .....	9
2.4.2. Sumber Daya/Power .....	9
2.4.3. Memori .....	9
2.4.4. Input / Output .....	10
2.4.5. Komunikasi .....	11
2.5. Raspberry Pi .....	11
2.6. Navigasi <i>Waypoint</i> .....	13
2.6.1. GPS ( <i>Global Positioning System</i> ) .....	15
2.6.2. Sensor GPS Ublox Neo M8N .....	16
2.6.3. Kompas Digital HMC5883L .....	18
2.6.4. <i>Heading</i> .....	18
2.7. Kendali Proporsional Integral Derivatif (PID) .....	20
2.7.1. Kendali Proporsional .....	21
2.7.2. Kendali Integral .....	21
2.7.3. Kendali Derivatif .....	21
2.8. Driver Motor BTN7960 .....	22

2.9. Photovoltaic (PV).....	22
2.10. Solar Charger Controller (SCC).....	23
2.11. Batterai Lipo ( <i>Lithium Polimer</i> ).....	24
2.12. Website .....	26
2.12.1. Database MySQL .....	26
2.12.2. <i>Hypertext Preprocessor</i> (PHP) .....	28
2.12.3. <i>Hypertext Markup Language</i> (HTML).....	29
BAB III. PERANCANGAN DAN REALISASI ALAT .....	31
3.1. Perancangan Sistem.....	31
3.2. Perancangan Perangkat Keras .....	34
3.2.1. Perancangan Sensor GPS Ublox Neo M8N .....	36
3.2.2. Perancangan Sensor Kompas .....	37
3.2.3. Perancangan Rangkaian <i>Driver Motor</i> BTN7960 ...	37
3.2.4. Perancangan Sensor Gas .....	33
3.2.5. Koneksi Raspberry Pi.....	40
3.3. Perancangan Perangkat Lunak .....	41
3.3.1. Desain Kendali PID Pada Differential Steering ....	41
3.3.2. Navigasi <i>Waypoint</i> .....	44
3.3.3. Mengakses Modul GPS Ublox Neo M8N .....	45
3.3.4. Mengakses Sensor Gas TGS2600 .....	47
3.3.5. Mengakses Driver Motor .....	48
3.4. Perancangan Catu Daya.....	48
3.4.1. Perhitungan Torsi .....	48
3.4.2. Perhitungan Daya .....	49
3.4.3. Perhitungan Daya Tahan Batterai .....	50
3.5. Perancangan Website.....	50
3.5.1. Konfigurasi IP <i>Address</i> Raspberry PI.....	51
3.5.2. Pembuatan Database MySQL PhpMyAdmin.....	51
3.5.3. Desain Website .....	52
BAB IV. ANALISA DAN PENGUJIAN SISTEM .....	55
4.1. Pengujian Sensor Kompas.....	56
4.2. Pengujian Sensor GPS Ublox Neo M8N.....	58
4.3. Pengujian Sensor Gas.....	59
4.4. Pengujian Panel Surya.....	61
4.5. Pengujian Batterai .....	63
4.6. Pengujian <i>Waypoint</i> .....	65
4.7. Pengujian <i>Uploading</i> Data Ke Database .....	69
4.8. Pengujian Sistem Secara Keseluruhan .....	71

BAB V. PENUTUP .....	75
5.1. Kesimpulan .....	75
5.2. Saran .....	75
DAFTAR PUSTAKA .....	77
LAMPIRAN A .....	81
LAMPIRAN B .....	83
LAMPIRAN C .....	91
LAMPIRAN D .....	93
LAMPIRAN E.....	101
BIODATA PENULIS.....	103

.....*Halaman ini sengaja dikosongkan*.....

## DAFTAR GAMBAR

Gambar 2.1. Bentuk penghalang potensial antar butir Kristal mikro SnO <sub>2</sub> saat tanpa adanya gas .....	6
Gambar 2.2. Bentuk penghalang potensial antar butir Kristal mikro SnO <sub>2</sub> saat lingkungan adanya gas .....	6
Gambar 2.3. Struktur Sensor Figaro TGS 2600 .....	7
Gambar 2.4. Board Arduino Mega 2560 .....	8
Gambar 2.5. Raspberry Pi 3 Model B .....	12
Gambar 2.6. Titik navigasi waypoint .....	14
Gambar 2.7. Penggambaran titik target terhadap objek.....	14
Gambar 2.8. Ilustrasi Satelit GPS .....	16
Gambar 2.9. GPS Ublox Neo M8N.....	17
Gambar 2.10 <i>Compass</i> HMC5883L .....	18
Gambar 2.11 <i>Eight Point Compass</i> .....	19
Gambar 2.12 <i>One Degree Compass</i> .....	19
Gambar 2.13 Blok diagram PID.....	20
Gambar 2.14 Motor <i>driver</i> BTN7960.....	22
Gambar 2.15 <i>Photovoltaic</i> (PV).....	23
Gambar 2.16 Konsep <i>Charge Controlle</i> .....	24
Gambar 2.17 Flexzion SCC PWM 20 A.....	24
Gambar 2.18 Baterai Lipo 4s 2600 mAh .....	25
Gambar 3.1. Blok diagram sistem. ....	32
Gambar 3.2. <i>Flowchart</i> Sistem.....	34
Gambar 3.3. Desain Robot Autodesk Inventor.....	35
Gambar 3.4. Blok diagram sistem elektronis.....	36
Gambar 3.5. <i>Schematic</i> GPS dengan Arduino Mega2560 .....	36
Gambar 3.6. <i>Schematic</i> kompas dengan Arduino Mega2560 .....	37
Gambar 3.7. Rangkaian <i>driver</i> motor dengan Arduino .....	38
Gambar 3.8. Skematik sensor gas dengan Arduino Mega2560 .....	38
Gambar 3.9. Grafik perbandingan Ro/Rs dengan kadar gas.....	39
Gambar 3.10. Grafik persamaan ppm gas iso-butana .....	40
Gambar 3.11. Antarmuka Raspberry Pi dengan Arduino. ....	41
Gambar 3.12. Diagram blok kendali PID .....	41
Gambar 3.13. Diagram alir algoritma PID .....	42
Gambar 3.14. Diagram alir algoritma <i>waypoint</i> . ....	45
Gambar 3.15. Ilustrasi perhitungan torsi .....	49
Gambar 3.16. a) Tampilan web menu home (PC) b) Tampilan web menu home ( <i>smartphone</i> ) .....	53
Gambar 3.17. Tampilan web menu peta (PC).....	54

Gambar 4.1. Realisasi <i>Mobile Robot</i> .....	55
Gambar 4.2. Lokasi pengujian pada Google Map.....	56
Gambar 4.3. Grafik pengujian sensor Kompas HMC4883L .....	57
Gambar 4.4. Pengujian sensor gas .....	60
Gambar 4.5. Nilai Ro pada saat udara bersih.....	60
Gambar 4.6. Grafik konsentrasi gas dalam ruangan. ....	61
Gambar 4.7. Grafik konsentrasi gas di lapangan. ....	61
Gambar 4.8. Pengujian panel surya .....	63
Gambar 4.9. Grafik pengukuran daya terhadap intensitas cahaya .....	64
Gambar 4.10. Grafik pengukuran tegangan terhadap arus.....	64
Gambar 4.11. Hasil plot <i>Waypoint</i> di Google Maps.....	67
Gambar 4.12. Hasil plotting koordinat yang dilalui robot satellite view .....	69
Gambar 4.13. Hasil plotting koordinat yang dilalui robot map view ..	69
Gambar 4.14. Program pengiriman data GPS dan Kadar gas di Arduino.....	70
Gambar 4.15. Program penerimaan data GPS dan Kadar gas di Python.....	72
Gambar 4.16. Serial LX Terminal Raspberry.....	72
Gambar 4.17. Penyimpanan database MySQL .....	73
Gambar 4.18. Pemetaan Kadar gas pada web tampilan PC .....	73
Gambar 4.19. Pemetaan Kadar gas pada web tampilan <i>Smartphone</i> ..	74
Gambar 4.20. Grafik distribusi hasil pemetaan konsentasi gas di lapangan .....	74

## DAFTAR TABEL

Tabel 2.1. Struktur Sensor Figaro TGS 2600 .....	7
Tabel 2.2. Spesifikasi Arduino Mega 2560 .....	9
Tabel 2.3. Spesifikasi U-Blox Neo-M8N GPS <i>compass</i> .....	17
Tabel 2.4. Spesifikasi PV .....	23
Tabel 2.5. Kelebihan dan Kekurangan Baterai Lipo .....	25
Tabel 3.1. Spesifikasi <i>Mobile Robot</i> .....	32
Tabel 3.2. Perbandingan Rs/Ro dengan Kadar gas .....	39
Tabel 4.1. Pengujian sensor kompas HMC4883L .....	57
Tabel 4.2. Tabel 4.2 Pengujian sensor GPS dengan GPS acuan .....	58
Tabel 4.3. Pengukuran daya maksimum PV .....	63
Tabel 4.4. Pengukuran penggunaan arus pada robot .....	64
Tabel 4.5. Pengukuran durasi <i>charging</i> baterai .....	64
Tabel 4.6. Pengukuran durasi <i>discharging</i> baterai .....	64
Tabel 4.7. Daftar <i>waypoint</i> .....	65
Tabel 4.8. Titik-titik koordinat <i>waypoint</i> yang dilalui robot .....	66
Tabel 4.9. Pemetaan konsentrasi gas .....	72

.....*Halaman ini sengaja dikosongkan*.....



## DAFTAR RUMUS

2.1. Persamaan magnitude <i>Mobile Robot</i> .....	15
2.2. Persamaan sudut arah <i>Mobile Robo</i> .....	15
3.1. Persamaan Hukum Newton II.....	48
3.2. Persamaan torsi roda.....	49
3.3. Persamaan daya .....	50
3.4. Persamaan Kapasitas Batterai.....	50

.....*Halaman ini sengaja dikosongkan*.....

# **BAB I**

## **PENDAHULUAN**

### **1.1. Latar Belakang**

Pemetaan kadar dan penyebaran gas berbahaya harus diketahui lokasinya agar dapat meminimalkan kerugian yang timbul karena pencemarannya, terutama tempat-tempat yang akan dilakukan eksploitasi tambang perminyakan [1]. Pemetaan kadar gas pada daerah-daerah yang akan dilakukan eksploitasi tambang perminyakan, terdapat beberapa gas berbahaya seperti karbon monoksida, gas karbondioksida, gas hidrogen sulfida, dan Liquid Petroleum Gas sangatlah berbahaya bagi keselamatan manusia [2].

Diperlukannya suatu alat yang dapat menggantikan peran manusia dalam melakukan pemetaan kadar gas-gas berbahaya pada setiap saat. Alat yang dapat digunakan untuk melakukan hal tersebut dan mengidentifikasi kemunculan dan penyebaran suatu gas berbahaya pada suatu tempat salah satunya menggunakan robot. Karena dengan menggunakan robot, pemetaan gas berbahaya tidak akan berpengaruh terhadap robot tersebut dan dapat berkerja setiap saat. Sehingga pengambilan data kadar gas berbahaya dapat dilakukan pada titik-titik lokasi yang berbeda, lebih luas dan *up to date*.

Dari metode sebelumnya, untuk pemetaan kadar gas dalam suatu peta dapat dilakukan dengan menggunakan Quadcopter [3,4]. Tetapi cara ini kurang efektif, karena pembacaan kadar gas dapat terganggu oleh kecepatan angin dari baling-baling quadcopter tersebut. Apalagi pemetaan kadar gas dengan menggunakan wireless network sensor [5,6], tentunya harus membutuhkan banyak sensor yang dipasang disetiap titik dari kadar gas yang akan dipetakan. Sedangkan pemetaan kadar gas dengan menggunakan balon udara [7,8], masih kurang efektif karena balon udara dalam pemetaan kadar gas sangat sulit disebabkan balon udara diam ditempat dan mudah terbawa oleh angin.

Oleh karena itu, pada penelitian ini dirancang suatu robot yaitu Mobile Robot Bertenaga Surya Untuk Pemetaan Konsentrasi Gas yang dilengkapi dengan *Global Position System* (GPS), sensor gas dan panel surya yang dapat berkerja setiap saat [9,10]. Data hasil pemetaan kadar gas berbahaya dari robot tersebut akan

ditampilkan dalam bentuk peta visual *website* yang dapat diakses melalui *internet*. Sehingga didapatkan kadar gas berbahaya pada titik-titik yang dilewati oleh robot tersebut dengan hanya satu sensor. Jalur lintasan *waypoint* robot dapat ditentukan di Google Map serta dilengkapi dengan panel surya sebagai catu dayanya dan adanya stasiun pengisian baterai via wireless apabila baterai robot tersebut habis di malam hari.

## **1.2. Perumusan Masalah**

Berdasarkan latar belakang di atas, dapat dirumuskan beberapa masalah. Antara lain :

- a. Jenis sensor apa yang digunakan untuk mendeteksi gas berbahaya.
- b. Bagaimana merancang sistem power supply yang diperlukan oleh mobile robot untuk investigasi di luar ruangan.
- c. Bagaimana cara untuk menampilkan hasil pemetaan agar dapat dipantau dari jarak jauh.

## **1.3. Tujuan Penelitian**

Penelitian tugas akhir ini bertujuan sebagai berikut :

- a. Mampu menerapkan sensor semikonduktor untuk mendeteksi gas berbahaya.
- b. Mampu membuat sistem Mobile Robot Bertenaga Surya Untuk Pemetaan Konsentrasi Gas yang dilengkapi dengan panel surya.
- c. Mampu menampilkan data hasil pemetaan konsentrasi gas dalam bentuk *website* yang terintegrasi dengan Google Map.

## **1.4. Batasan Masalah**

Batasan masalah dari tugas akhir ini adalah sebagai berikut:

- a. Tidak adanya hujan atau badai pada saat robot sedang beroperasi sesuai lokasi dan waktu yang ditentukan.
- b. Tidak adanya angin yang besar.
- c. Tidak adanya tanjakan atau jalan berlubang yang dilalui oleh robot.
- d. Lokasi tempat beroperasi dapat terhubung dengan internet.
- e. Sensor yang digunakan adalah sensor TGS2600 untuk mendeteksi konsentrasi gas iso butana.

### 1.5. Metodologi Penelitian

Metodologi yang digunakan pada penelitian tugas akhir ini adalah sebagai berikut :

- a. Studi literatur tentang mobile robot bertenaga surya.

Pada bagian ini mempelajari buku-buku *literature* yang digunakan untuk pengumpulan dasar teori dan metode-metode apa yang digunakan untuk menunjang dalam pengerjaan tugas akhir. Dasar teori tersebut diambil dari jurnal dan buku-buku penunjang. Berikut adalah refrensi teori yang berisi:

1. Navigasi waypoint
2. Data Logger Raspberry Pi
3. Kontrol PID

- b. Perancangan perangkat keras

Pada tahap ini penuangan konsep dan desain untuk mekanis, elektronis maupun software, agar mendapatkan kesempurnaan alat. Desain mekanik robot dimulai dengan perancangan perangkat keras sistem kendali yaitu penggabungan fungsi beberapa komponen seperti Panel surya, sensor-sensor, mikrokontroller, driver motor, antena dan aktuator.

- c. Perancangan Sistem mobile robot

Sistem yang akan dirancang meliputi integrasi modul-modul sensor, driver motor, interface data logger dan sistem catu daya panel surya.

- d. Pengujian alat

Pada tahap ini, dilakukan untuk menguji rangkaian dan perancangan mekanik yang dirancang sesuai dengan yang diharapkan. Pengujian dilakukan pada *software* dan *hardware* yang telah dibuat untuk menguji keakuratan alat dengan menjalankannya pada lingkungan terbuka, disertai memberikan beberapa titik lokasi yang diberi gas.

- e. Penulisan laporan tugas akhir

Tahap penulisan laporan tugas akhir dilakukan pada saat pengujian sistem dimulai dan setelahnya.

## **1.6. Sistematika penulisan**

Laporan tugas akhir ini terdiri dari lima bab dengan sistematika penulisan sebagai berikut :

a. Bab 1 : Pendahuluan

Bab ini meliputi latar belakang, perumusan masalah, tujuan penelitian, batasan masalah, metodologi, sistematika penulisan dan relevansi.

b. Bab 2 : Dasar Teori

Bab ini menjelaskan tentang dasar-dasar teori tentang Sensor semikonduktor, modul kompas digital, GPS, sistem navigasi, papan mikrokontroller, serial komunikasi data, internet data logging dan antar muka perangkat lunak pada Raspberry Pi.

c. Bab 3 : Perancangan Alat

Bab ini menjelaskan tentang perancangan perangkat keras dan perangkat lunak untuk pengaplikasian penelitian.

d. Bab 4 : Pengujian Alat

Bab ini menjelaskan tentang hasil yang didapat dalam pengujian dari tiap blok sistem dan subsistem serta hasil evaluasi sistem tersebut.

e. Bab 5 : Penutup

Bab ini menjelaskan tentang kesimpulan meliputi hasil yang dicapai, dan kekurangan-kekurangan pada kerja alat dari hasil analisa serta saran untuk pengembangan ke depan.

## **1.7. Relevansi**

Hasil dari tugas akhir ini diharapkan dapat memberikan manfaat sebagai berikut:

- a. Dapat mendukung penelitian khususnya pada bidang studi elektronika tentang penggunaan sensor gas dalam aplikasi pengukuran dan penelitian selanjutnya.
- b. Mendukung penelitian tentang robotika dengan navigasi sistem otomatis yang dapat dimanfaatkan untuk keperluan lebih luas.
- c. Dihasilkan aplikasi yang terdapat terhubung dengan internet untuk mendukung penelitian tentang pemantauan, pendeteksian dan pemetaan kadar gas pada ruang terbuka.

## **BAB II**

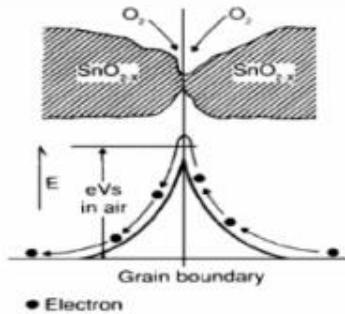
### **DASAR TEORI**

#### **2.1. Sensor Semikonduktor**

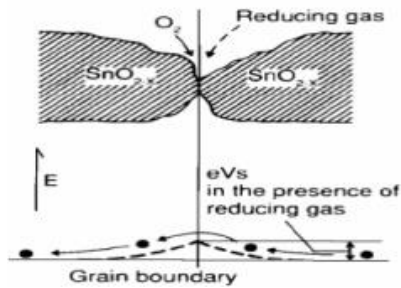
Sensor gas adalah sensor yang berfungsi untuk mengukur senyawa gas polutan yang ada di udara seperti karbon monoksida, hidrokarbon, nitrooksida, dan lain-lain. Sensor gas semikonduktor memiliki banyak jenis yang dibedakan oleh sensitivitas sensor tersebut. Prinsip kerja dari sensor gas semikonduktor ini ialah ketika mendeteksi adanya perubahan konsentrasi gas di udara sesuai dengan target gas yang dituju maka resistansi dari sensor akan semakin rendah, dan akan berangsur-angsur naik ketika konsentrasi gas di udara sekitar sensor gas semikonduktor mulai turun secara perlahan [8].

#### **2.2. Prinsip Kerja Sensor Semikonduktor**

Sensor gas tipe semikonduktor terdiri dari elemen sensor, dasar sensor dan tudung sensor. Elemen sensor menggunakan bahan-bahan seperti timah(IV)oksida  $\text{SnO}_2$ , wolfram (VI) oksida  $\text{WO}_3$ , dan lain-lain. Bila suatu kristal oksida logam seperti  $\text{SnO}_2$  dipanaskan pada suhu tinggi tertentu di udara, oksigen teradopsi pada permukaan kristal dengan muatan negatif. Elektron – elektron donor pada permukaan kristal ditransfer ke oksigen teradopsi, sehingga menghasilkan suatu lapisan ruang bermuatan positif. Akibat dari peristiwa tersebut potensial permukaan terbentuk yang akan menghambat aliran elektron. Proses didalam sensor arus listrik mengalir melalui bagian – bagian penghubung (batas butir) kristalkristal mikro  $\text{SnO}_2$ . Di batas-batas antar butir, oksigen yang teradopsi membentuk penghalang potensial yang menghambat muatan bebas bergerak.



Gambar 2.1 Bentuk penghalang potensial antar butir Kristal mikro  $\text{SnO}_2$  saat tanpa adanya gas [8]



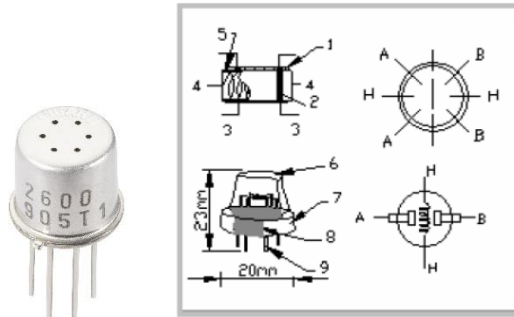
Gambar 2.2 Bentuk penghalang potensial antar butir Kristal mikro  $\text{SnO}_2$  saat lingkungan gas [8]

Ketika model penghalang potensial antar butir dalam lingkungan gas atau terdeteksi gas, kerapatan oksigen teradsorpsi bermuatan negatif pada permukaan semikonduktor sensor menjadi berkurang, sehingga ketinggian penghalang pada batas antar butir berkurang yang menyebabkan berkurangnya tahanan butir dalam lingkungan gas.

### 2.3. Sensor Gas TGS 2600

Sensor Gas Figaro TGS 2600 merupakan sebuah sensor kimia atau gas sensor. Sensor ini mempunyai nilai resistansi  $R_s$  yang akan berubah bila terkena gas dan juga mempunyai sebuah pemanas (heater) yang digunakan untuk membersihkan ruangan sensor dari kontaminasi udara luar [11]. Bahan detektor gas dari sensor adalah metal oksida, khususnya senyawa  $\text{SnO}_2$ .





Gambar 2.3 Struktur Sensor Figaro TGS 2600[8]

Tabel 2.1 Struktur Sensor Figaro TGS 2600 [8]

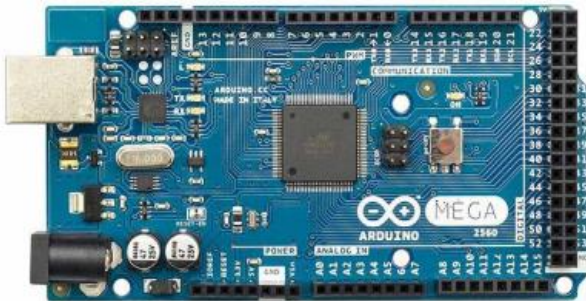
	Parts	Materials
1	Gas sensing layer	$\text{SnO}_2$
2	Electrode	Au
3	Electrode line	Pt
4	Heater coil	Ni-Cr alloy
5	Tubular ceramic	$\text{Al}_2\text{O}_3$
6	Anti-explosion network	Stainless steel gauze (SUS316 100-mesh)
7	Clamp ring	Copper plating Ni
8	Resin base	Bakelite
9	Tube Pin	Copper plating Ni

Ketika kristal metal oksida ( $\text{SnO}_2$ ) dihangatkan pada temperatur tertentu, oksigen akan diserap pada permukaan kristal dan oksigen akan bermuatan negatif. Hal ini disebabkan karena permukaan kristal mendonorkan elektron pada oksigen yang terdapat pada lapisan luar, sehingga oksigen akan bermuatan negatif dan muatan positif akan terbentuk pada permukaan luar kristal. Tegangan permukaan yang terbentuk akan menghambat laju aliran elektron [7].

## 2.4. Arduino Mega 2560

Arduino Mega 2560 adalah papan pengembangan mikrokontroller yang berbasis Arduino dengan menggunakan chip ATmega2560 seperti pada Gambar 2.4. *Board* ini memiliki pin I/O yang cukup banyak, sejumlah 54 buah digital I/O pin (15 pin diantaranya adalah PWM), 16 pin analog input, 4 pin UART (serial port hardware). Arduino Mega 2560 dilengkapi dengan sebuah oscillator 16 Mhz, sebuah port USB, power jack DC, ICSP header, dan tombol reset. Board ini sudah sangat lengkap, sudah memiliki segala sesuatu yang dibutuhkan untuk sebuah mikrokontroller. Dengan penggunaan yang cukup sederhana, anda tinggal menghubungkan power dari USB ke PC anda atau melalui adaptor AC/DC ke jack DC [8].

Sifat open source arduino juga banyak memberikan keuntungan tersendiri untuk dalam menggunakan board ini, karena dengan sifat open source komponen yang dipakai tidak hanya tergantung pada satu merek, namun memungkinkan untuk bisa memakai semua komponen yang ada dipasaran. Bahasa pemrograman arduino merupakan bahasa C yang sudah disederhanakan *syntax* bahasa pemrogramannya sehingga mempermudah dalam mempelajarinya.



Gambar 2.4. Board Arduino Mega 2560 [2]

Tabel 2.2 Spesifikasi Arduino Mega 2560 [2]

No	Parameter	Spesifikasi
1	Mikrokontroler	Atmega2560
2	Tegangan Operasi	5 V
3	Tegangan Input (Rekomendasi)	7-12 V
4	Tegangan Input (Batas)	6-20 V
5	Pin I/O Digital	54 (14 Pin PWM)
6	Pin Input Analog	16
7	Arus DC per pin I/O	40 mA
8	Arus DC untuk pin 3.3 V	50 mA
9	Flash Memory	256 KB (8 KB digunakan <i>bootloader</i> )
10	SRAM	8 KB
11	EEPROM	4 KB
12	Clock Speed	16 MHz

#### 2.4.1. Spesifikasi Arduino Mega 2560

Spesifikasi dari Arduino Mega 2560 ditunjukkan pada tabel 2.2.

#### 2.4.2. Sumber Daya/Power

Arduino Mega 2560 dapat diaktifkan melalui kabel USB atau dengan suplai daya eksternal. *Board* dapat beroperasi pada suplai eksternal dari 6 hingga 20 volt. Rentang tegangan operasi yang disarankan adalah 7 hingga 12 volt karena jika tegangan kurang dari 6 volt maka tidak akan stabil, sedangkan jika lebih dari 12 volt, maka dapat menyebabkan regulator tegangan menjadi panas dan kemungkinan paling buruk adalah dapat merusak *board* Arduino Mega.

#### 2.4.3. Memori

Chip Atmega2560 yang terpasang pada Arduino Mega 2560 Revisi 3 memiliki memori 256 KB, dengan 8 KB dari memori tersebut telah digunakan untuk *bootloader*. Jumlah SRAM 8 KB dan EEPROM 4 KB, dimana pada memori ini dapat dilakukan baca-tulis dengan menggunakan EEPROM *library* saat melakukan pemrograman.

#### 2.4.4. *Input / Output*

Pin *input/output* Arduino Mega 2560 jumlahnya yang terbanyak dari semua jenis *board* yang dikembangkan Arduino. Arduino Mega 2560 memiliki 54 pin digital yang dapat digunakan sebagai *input/output*. Pin-pin tersebut bekerja pada tegangan 5 volt. Pada Arduino Mega terdapat beberapa pin yang memiliki fungsi khusus, yaitu:

- a. Serial: Disini terdapat 4 serial yang masing-masing terdiri dari 2 pin, yakni RX dan TX. (a) Serial 0: pin 0 (RX) dan pin 1 (TX), (b) Serial 1: pin 19 (RX) dan pin 18 (TX), (c) Serial 2: pin 17 (RX) dan pin 16 (TX), (d) Serial 3: pin 15 (RX) dan pin 14 (TX). Pin RX adalah pin untuk *receive* atau menerima, sedangkan TX adalah *transmit* atau untuk mengirim data serial TTL. Pin 0 dan pin 1 adalah pin yang digunakan oleh chip USB-to-TTL Atmega16U2.
- b. Eksternal *Interrupt*: Pin *interrupt* yaitu pin 2 (*interrupt* 0), pin 3 (*interrupt* 1), pin 18 (*interrupt* 5), pin 19 (*interrupt* 4), pin 20 (*interrupt* 3), dan pin 21 (*interrupt* 2). Pin-pin tersebut dapat dikonfigurasi untuk men-*trigger* sebuah *interrupt* pada nilai “LOW”, tepi naik atau tepi turun, atau merubah nilai. Untuk mengatur *interrupt* tersebut, maka perlu menggunakan fungsi *attachInterrupt()*.
- c. PWM: Pin 2 hingga 13 dan 44 hingga 46 adalah pin-pin yang menyediakan *output* PWM 8-bit dengan menggunakan fungsi *analogWrite()*.
- d. SPI: Pin 50 (MISO), 51 (MOSI), 52 (SCK), dan 53 (SS) mendukung komunikasi SPI dengan menggunakan *SPI library*.
- e. LED: Pin 13, pin ini terhubung dengan *built-in* LED yang dikendalikan oleh pin digital nomor 13. Set HIGH untuk menyalakan LED, LOW untuk memadamkannya.
- f. I2C: Pin 20 (SDA) dan Pin 21 (SCL). Pin-pin ini mendukung komunikasi I2C (TWI) menggunakan *wire library*.

Selain pin-pin *digital*, juga terdapat pin-pin *analog* yang berjumlah 16 buah. Masing-masing pin *analog* tersebut memiliki resolusi 10 bits (rentang nilai dari 0 sampai 1024). Secara *default*, pin-pin ini diukur dari *ground* ke 5 volt, namun bisa juga dengan menggunakan fungsi *analogReference()*. Beberapa pin lainnya pada *board* ini yaitu:

- a. AREF: sebagai referensi tegangan *input analog*.
- b. Reset: Dengan menghubungkan ke “LOW” untuk melakukan reset terhadap mikrokontroler atau bisa juga dengan menekan tombol reset yang tersedia.

#### 2.4.5. Komunikasi

Arduino Mega 2560 dilengkapi fasilitas untuk berkomunikasi dengan komputer, arduino lainnya, ataupun dengan mikrokontroler lainnya. Chip Atmega 2560 menyediakan komunikasi serial UART TTL (5V) yang tersedia pada pin 0 (RX) dan pin 1 (TX). Chip Atmega16U2 pada *board* berfungsi untuk menerjemahkan bentuk komunikasi melalui USB dan akan tampil sebagai *virtual port* di komputer. *Firmware* 16U2 menggunakan driver USB standar sehingga tidak memerlukan driver tambahan.

Pada Arduino *Software* (IDE) terdapat serial monitor yang memudahkan data tekstual untuk dikirim menuju Arduino atau keluar dari Arduino. Ketika ada data yang ditransmisikan melalui Chip USB to serial via kabel USB ke komputer maka LED TX dan RX akan menyala berkedip-kedip. Pin digital juga dapat digunakan untuk komunikasi serial, yaitu dengan menggunakan *library SoftwareSerial*. Selain itu, Atmega2560 juga mendukung komunikasi I2C (TWI) dan SPI. Pada Arduino *Software* (IDE) telah terdapat *wire library* yang memudahkan pengguna menggunakan bus I2C, serta *SPI library* yang dipakai untuk menggunakan komunikasi SPI.

#### 2.5. Raspberry Pi

*Raspberry pi* dapat digunakan untuk banyak hal seperti yang komputer bisa lakukan, seperti *spreadsheets*, *word processing*, permainan, dan juga pemrograman. Selain itu *Raspberry pi* digunakan untuk pengontrolan lebih dari satu *device*, baik jarak dekat ataupun jarak jauh. Berbeda dengan mikrokontroler, *Raspberry pi* dapat mengontrol lebih dari 1 unit *device* yang ingin dikontrol. Untuk pengontrolan unit *device* yang akan dikontrol, *Raspberry pi* menggunakan bahasa Python sebagai bahasa pemrogramannya. *Raspberry pi* memiliki beberapa berbagai fitur, yaitu *Micro SD* yang berfungsi sebagai *harddisk*, *port usb*, *port Ethernet*, *audio video output*, HDMI Video, CPU 400-700 MHz, dan yang paling penting adalah *Raspberry pi* memiliki pin GPIO yang berfungsi untuk *interface* dengan berbagai perangkat elektronik[8]. Bahasa yang digunakan dalam pengontrolan adalah

bahasa Python. Bahasa python adalah bahasa pemrograman yang memiliki banyak fungsi, interaktif, berorientasi objek dan merupakan bahasa pemrograman tingkat tinggi. Bahasa python adalah bahasa pemrograman formal dengan aturan-aturan dan format spesifiknya sendiri.



Gambar 2.5. Raspberry Pi 3 Model B [7]

*Raspbian* adalah sistem operasi *Linux distro Debian* yang dioptimalkan untuk penggunaan perangkat komputer mini *raspberry pi*. Sistem operasi ini memiliki beberapa program standard dan beberapa alat bantu untuk dapat menjalankan perangkat keras dari komputer mini *Raspberry Pi* [12]. Selain *Raspbian*, sistem operasi yang dapat digunakan pada *Raspberry pi* pada umumnya *raspbian wheezy* dan sistem ini yang paling banyak digunakan oleh para pengguna *Raspberry pi*.

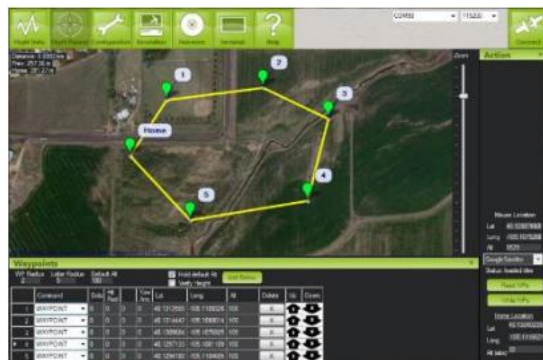
*Raspian stretch* adalah sistem operasi gratis yang berbasis Debian yang telah dioptimalkan untuk *Raspberry pi*, Agar GPIO (*General Input Output*) pada *Raspberry pi* bekerja dengan baik maka terlebih dahulu menginstal *library* dari BCM2835 pada *Raspberry pi*. *Raspberry pi* adalah pin *generic* pada *chip* yang dapat dikontrol (diprogram) melalui perangkat lunak baik di konfigurasi sebagai pin input maupun pin output. *Raspberry pi* GPIO memiliki 40 pin dengan ukuran 2,54 mm. Konektor GPIO memiliki fitur-fitur diantaranya [13] :

- a. Pin antarmuka I2C yang memungkinkan untuk menghubungkan modul *hardware* dengan hanya dua pin kontrol.
- b. SPI antarmuka, memiliki konsep mirip dengan I2C tetapi dengan standar yang berbeda.
- c. Serial Rx dan Tx, pin untuk berkomunikasi dengan perangkat serial.
- d. Pin PWM (*Pulse Width Modulation*) untuk kontrol daya.
- e. Pin PPM (*Pulse Position Modulation*) untuk mengendalikan motor servo.

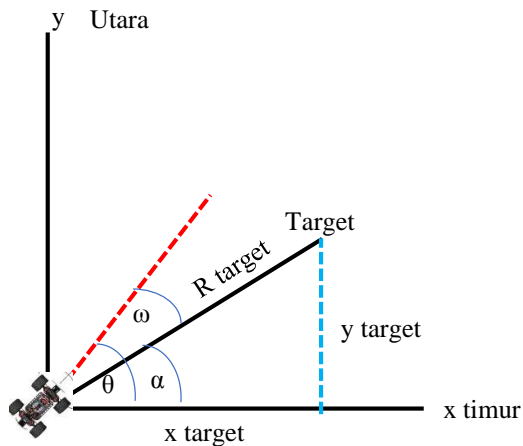
## **2.6. Navigasi Waypoint**

Navigasi merupakan teknik untuk membaca kedudukan (posisi) dan arah benda terhadap kondisi di sekitarnya. Pada umumnya menemukan kedudukan dan posisi dari benda dapat dilakukan dengan melihat beberapa penanda dari sensor [15]. Metode umum untuk mewujudkan navigasi otomatis dapat melalui navigasi visual, model peta navigasi dan navigasi waypoint [16].

Koordinat yang digunakan dapat bervariasi tergantung pada aplikasi. Misalnya yang berkenaan dengan navigasi pada permukaan bumi, koordinat ini dapat mencakup latitude dan longitude [17]. Waypoint memiliki cakupan yang luas dalam bidang navigasi oleh orang awam, semenjak dikembangkan sistem navigasi lanjutan. Waypoint yang ditempatkan di atas permukaan bumi, biasanya didefinisikan dalam dua dimensi (latitude dan longitude) [18]. Pada navigasi waypoint dalam dua dimensi, penentuan posisi target terhadap suatu objek atau benda dapat dilihat pada Gambar 2.6 dan Gambar 2.7.



Gambar 2.6. Titik navigasi waypoint[4]



Gambar 2.7. Penggambaran titik target terhadap objek[2]



Pada Gambar 2.7, dapat dilihat bahwa jarak antara posisi benda dengan target dapat dihitung dengan persamaan (2.1).

$$R_{\text{target}} = \sqrt{X_{\text{target}}^2 + Y_{\text{target}}^2} \quad (2.1)$$

Variable  $X_{\text{target}}$  dan  $Y_{\text{target}}$  merupakan koordinat latitude dan longitude yang akan dituju. Sehingga arah yang harus dituju oleh benda ( $\omega$ ) dapat dihitung dengan persamaan (2.2).

$$\alpha = \tan^{-1}\left(\frac{Y_{\text{target}}}{X_{\text{target}}}\right)$$

$$\omega = \theta - \alpha \quad (2.2)$$

Sudut  $\alpha$  pada persamaan 2 merupakan sudut yang dibentuk setelah menentukan titik target, dan sudut  $\theta$  merupakan sudut yang didapat dari kompas.

### 2.6.1. GPS (*Global Positioning System*)

Merupakan sistem navigasi yang menggunakan satelit. Didesain dapat memberikan informasi posisi secara mudah, dengan posisi dapat dikenali hampir semua di permukaan bumi [2]. Sistem navigasi merupakan bagian terpenting, terutama untuk mengetahui posisi suatu objek tanpa harus memerhatikan lingkungan sekitar. GPS memberikan informasi lokasi dengan koordinat lintang (*latitude*) dan bujur (*longitude*) berdasarkan koordinat bumi

Untuk dapat mengetahui posisi seseorang maka diperlukan alat yang diberinama GPS reciever yang berfungsi untuk menerima sinyal yang dikirim dari satelit GPS. Posisi di ubah menjadi titik yang dikenal dengan nama *Way-point* nantinya akan berupa titik-titik koordinat lintang dan bujur dari posisi seseorang atau suatu lokasi kemudian di layar pada peta elektronik. Sejak tahun 1980, layanan GPS yang dulunya hanya untuk keperluan militer mulai terbuka untuk publik. Uniknya, walau satelit-satelit tersebut berharga ratusan juta dolar, namun setiap orang dapat menggunakannya dengan gratis.



Gambar 2.8. Ilustrasi Satelit GPS [15]

Satelit-satelit ini mengorbit pada ketinggian sekitar 12.000 mil dari permukaan bumi. Posisi ini sangat ideal karena satelit dapat menjangkau area *coverage* yang lebih luas. Satelit-satelit ini akan selalu berada posisi yang bisa menjangkau semua area di atas permukaan bumi sehingga dapat meminimalkan terjadinya *blank spot* (area yang tidak terjangkau oleh satelit). Gambar 2.8, merupakan ilustrasi dari banyaknya satelit yang beredar di bumi.

Setiap mampu mengelilingi bumi hanya dalam waktu 12 jam. Sangat cepat, sehingga mereka selalu bisa menjangkau dimana pun posisi semua orang di atas permukaan bumi. GPS receiver sendiri berisi beberapa *integrated circuit* (IC) sehingga murah dan teknologinya mudah untuk di gunakan oleh semua orang. GPS dapat digunakan untuk berbagai kepentingan, misalnya mobil, kapal, pesawat terbang, pertanian dan di integrasikan dengan komputer maupun laptop.

#### 2.6.2. Sesnsor GPS Ublox Neo M8N

GPS sebagai satu-satunya satelit navigasi global yang dapat menentukan lokasi, kecepatan, arah, dan waktu adalah alat yang banyak digunakan di bumi ini [19]. Salah satu model GPS yang ada yakni GPS U-Blox Neo M8N ditunjukkan oleh Gambar 2.13.

Modul GPS NEO-M8 didesain untuk menerima dan melacak sinyal L1C/A pada frekuensi 1575.42 MHz dari GPS.

Seri NEO-M8N menghasilkan sinyal dengan sensitivitas tinggi dan waktu minimum akuisisi meskipun pada daya rendah sistem. GPS ini dapat disuplai dengan tegangan 3.3 hingga 5 volt. Spesifikasi lebih lengkap GPS NEO-M8N disajikan dalam tabel 2.5[20].



Gambar 2.9 GPS Ublox Neo M8N [20]

Tabel 2.3 Spesifikasi U-Blox Neo-M8N GPS *compass* [21]

Suplai	2.7 hingga 3.6 Volt (disarankan)
Antarmuka	UART, USB, SPI, dan DDC (I2C compliant)
Fitur	<i>Programmable (Flash)</i>
	<i>Data Logging</i>
	<i>Additional SAW</i>
	<i>Additional UNA</i>
	<i>RTC Crystal</i>



Gambar 2.10 *Compass HMC5883L* [22]

### 2.6.3. Kompas Digital HMC5883L

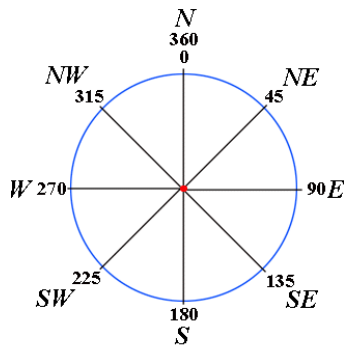
Kompas adalah alat yang digunakan untuk menentukan arah, mana selatan, mana utara, mana timur, dan mana barat. Kompas yang umum digunakan yaitu kompas analog [23]. Akan tetapi, selain kompas analog, juga ada kompas digital yang umum digunakan pada alat-alat elektronik. Salah satunya adalah kompas digital HMC5883L yang ditunjukkan oleh Gambar 2.10.

Kompas digital HMC5883L ini merupakan *magnetometer* yang digunakan untuk menghitung kuat medan magnet dan menghitung arahnya. HMC5883L, didalamnya termasuk *high-resolution HMC118X series magneto-resistive sensors* yang dilengkapi penguat, pembatalan *offset*, dan 12-bit ADC dengan akurasi sudut  $1^\circ$  hingga  $2^\circ$ . Antarmuka kompas ini menggunakan I2C. Tegangan suplai berada pada rentang 3 hingga 5 volt (disarankan 3.3 volt) dan rentang pengukuran berada di antara 1.3 hingga 8 Gauss [24].

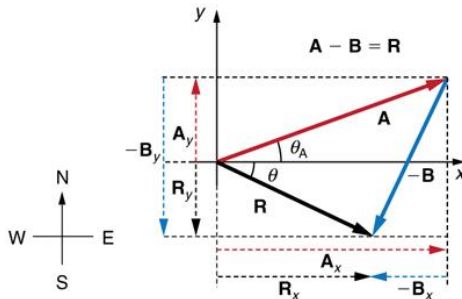
### 2.6.4. Heading

Ada dua bentuk metode kompas yang digunakan dalam sistem navigasi menggunakan sensor magnetoresistif adalah *the eight point compass* dan *the one-degree compass*. *Eight point compass* merupakan metode *heading* menggunakan penggambaran sederhana delapan titik kardinal kompas (U,S,B,T) dan titik tengah (TL, TG, BD, BL) seperti yang ditunjukkan pada Gambar 2.11.

Tipe kompas ini banyak digunakan dalam otomotif sederhana dimana pengendara ingin mengetahui arah perjalanan secara umum. Untuk aplikasi ini, sensor magnetik dapat direduksi hanya menjadi dua sumbu yaitu X dan Y. Dalam menganalisa respon sensor ditunjukkan pada Gambar 2.11. Dengan mengetahui medan magnet bumi selalu menunjuk ke arah utara, mulai analisa dengan sumbu X (perjalanan) lurus menuju utara. Keluaran sumbu X akan berada pada nilai maksimumnya ketika keluaran Y berada pada 0.



Gambar 2.11 *Eight Point Compass* [2]



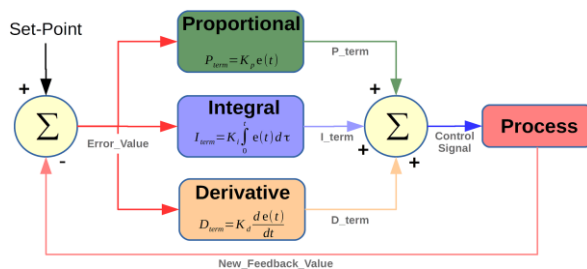
Gambar 2.12 *One Degree Compass* [2]

Kedua adalah *One Degree Compass* merupakan metode heading yang mencapai ketelitian  $1^\circ$ . Kompas membutuhkan sensor magnet yang dapat secara nyata memperbaiki perubahan *angular* hingga  $0.1^\circ$ . Sensor juga harus menunjukkan histerisis yang rendah ( $<0.05\%FS$ ), linearitas derajat yang tinggi ( $<0.05\%FS$  error) dan dapat dipakai berulang-ulang (konsiten). Medan magnet pada bidang X dan Y biasanya dapat mencapai sekitar 200 hingga 300 miligauss pada bagian ekuator dan kurang pada bagian sumbu. Sehingga berlaku hubungan pada persamaan. Secara vektor ditunjukkan pada Gambar.

## 2.7. Kendali Proporsional Integral Derivatif (PID)

Kendali proposional integral derivatif merupakan suatu sistem kendali gabungan dari kendali proporsional, kendali integral dan kendali drivatif. Dimana kendali proporsional akan menghasilkan keluaran kendali yang sebanding dengan nilai error, sehingga diperoleh nilai *steady state*. Kendali Integral berfungsi untuk meminimalisir nilai error *steady state* terhadap set point.

Untuk menghilangkan osilasi atau *overshoot* respon, maka ditambahkan kendali derivatif yang merupakan fungsi derivatif dari nilai error dikalikan dengan konstanta derivatif. Blok diagram kendali PID ditunjukkan pada Gambar 2.12. Dari setiap kendali yang menyusun kendali PID mempunyai karakteristik masing-masing.



Gambar 2.13 Blok diagram PID [2]

### 2.7.1. Kendali Proposional

Kendali proposional adalah kendali yang menghasilkan keluaran kendali yang sebanding dengan *error* masukan yang dikalikan dengan konstanta proposional. Kendali ini berfungsi untuk memperkuat sinyal *error*, sehingga mempercepat keluaran sistem mencapai titik referensi. Persamaan kendali proposional ini ditunjukkan pada persamaan

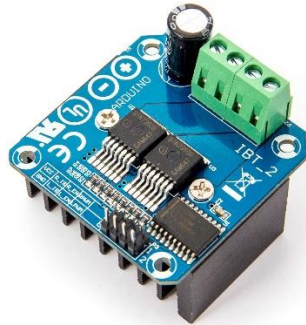
### 2.7.2. Kendali Integral

Kendali integral merupakan kendali yang menghasilkan keluaran yang sebanding lurus dengan besar dan lamanya *error*. Integral dalam kendali PID adalah jumlah *error* tiap waktu dan mengakumulasi *offset* yang sebelumnya telah dikoreksi. *Error* terakumulasi dikalikan dengan gain integral ( $K_i$ ) dan menjadi keluaran kendali. Persamaan kendali integral ini ditunjukkan pada persamaan.

Kendali integral mempercepat perpindahan proses menuju *setpoint* dengan menghilangkan *error stady state* yang muncul pada kendali proposional. Tetapi, karena integral mengakumulasi *error* sebelumnya maka dapat menyebabkan *overshoot*.

### 2.7.3. Kendali Derivatif

Kendali derivatif merupakan kendali yang menghasilkan keluaran sebanding dengan nilai selisih dari masukan. Kendali derivatif digabungkan dengan kendali proposional, akan meredam nilai *overshoot* dari keluaran kendali. Persamaan kendali derivatif ditunjukkan pada persamaan.



Gambar 2.14 Motor driver BTN7960 [2]

## 2.8. Driver Motor BTN7960

Merupakan *integrated high current half bridge* untuk aplikasi driver motor. *Driver* ini berisi satu p-channel MOSFET pada *highside* dan satu n-channel pada *lowside* yang terintegrasi menjadi satu. Antarmuka dengan mikrokontroler dibuat mudah oleh produsen dengan fitur *logic level input*, sensor arus, *slew rate adjustment*, penghasil waktu mati dan proteksi pada kelebihan temperature, arus, tegangan, hubung pendek, dan kurang tegangan. Tegangan *supply* dari -0.3 V hingga 45 V, tegangan *logic* dari -0.3 V hingga 5.5 V dengan arus beban maksimal 43 A.

## 2.9. Photovoltaic (PV)

*Photovoltaic* (PV) atau biasa dikenal sebagai solar panel ataupun panel surya adalah bahan semikonduktor yang berfungsi untuk konversi cahaya menjadi listrik secara langsung. Beberapa bahan menunjukkan sebuah properti yang biasa dikenal sebagai efek fotolistrik yang menyebabkan beberapa bahan tersebut dapat menyerap foton cahaya dan melepaskan elektron. Ketika elektron bebas ini ditangkap, maka akan dihasilkan listrik yang dapat digunakan sebagai sumber listrik. Total daya keluaran PV sebanding dengan tegangan operasi dikalikan dengan arus operasi PV. Setiap PV memiliki *Maximum Power Point* atau biasa dikenal sebagai titik operasi. PV yang digunakan ditunjukkan pada Gambar 2.15.





Gambar 2.15 Photovoltaic (PV)

Tabel 2.4. Spesifikasi PV

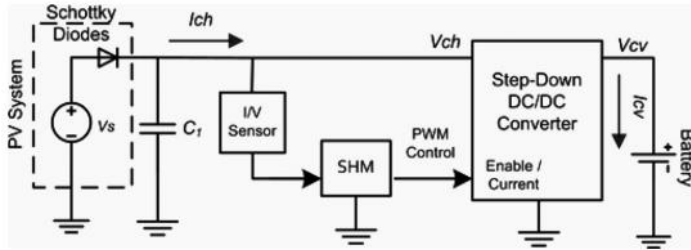
Makasimum Power	20 W
Tegangan operasi	17.2 V
Arus Operasi	1.18 A
Tegangan Rangkaian Terbuka (Voc)	21.6 V
Arus Rangkaian Tertutup (Isc)	1.23 A

## 2.10. Solar Charge Controller (SCC)

Pemanfaatan PV dengan dipasangkan baterai selalu menggunakan sistem *charge controller*. Tujuannya agar penggunaan baterai dan energi listrik dapat optimal dan aman. Yakni, pada pengisian berlebih dari PV, pemakaian berlebih oleh beban, mencegah arus balik ke PV, dan mencegah hubung singkat pada beban. Gambar 2.16 menunjukkan konsep dasar *charge controller* [25].

Tipe *Charge Controller* yang digunakan pada tugas akhir ini adalah Flexzion SCC PWM 20 A yang ditunjukkan pada gambar 2.17. Solar charge controller mengatur overcharging (kelebihan pengisian - karena batere sudah 'penuh') dan kelebihan voltase dari panel surya / *solar cell*. Kelebihan voltase dan pengisian akan mengurangi umur baterai. *Solar charge controller* menerapkan teknologi *Pulse Width Modulation* (PWM) untuk mengatur fungsi pengisian baterai dan pembebasan arus dari baterai ke beban.

Konektor yang berada pada kontroler ada tiga pasang, pertama untuk masukan daya dari solar panel, kedua keluaran untuk mengisi daya baterai, dan ketiga keluaran menuju beban yang akan disuplai.



Gambar 2.16 Konsep *Charge Controlle* [9]



Gambar 2.17 Flexzion SCC PWM 20 A [26]

## 2.11. Baterai LIPO (*Litium Polimer*)

Baterai Lipo sudah umum digunakan sebagai sumber pada sebuah rangkaian elektronika. Didalamnya terdapat elektrolit polimer kering yang berbentuk seperti lapisan plastik film yang tipis. Lapisan film ini disusun berlapis-lapis diantara anoda dan katoda sehingga dapat mengakibatkan terjadinya pertukaran ion. Metode ini memungkinkan Lipo untuk dibuat dalam berbagai bentuk dan ukuran. Namun, karena terdapat sekat yang berupa elektrolit polimer kering yang membatasi antara anoda dan katoda membuat aliran pertukaran ion menjadi lemah. Hal ini berakibat pada penurunan *charging* dan *discharging rate*. Solusi untuk masalah ini yaitu dengan memanaskan baterai, sehingga pertukaran ion akan menjadi lebih cepat. Akan tetapi, masalah keamananlah yang berbicara, dimana memanaskan baterai dapat membuat baterai terbakar atau bahkan meledak. Baterai Lipo memiliki kelebihan sekaligus kekurangan jika dibandingkan baterai

jenis lain seperti NiCd atau NiMH seperti yang disebutkan dalam tabel 2.5. Pada penelitian ini, menggunakan baterai lipo 4 sel yang ditunjukkan pada Gambar 2.16.



Gambar 2.18 Baterai Lipo 4s 2600 mAh [27]

Tabel 2.5. Kelebihan dan Kekurangan Baterai Lipo [28].

Kelebihan	Kekurangan
Bobot ringan dan tersedia dalam berbagai bentuk dan ukuran.	Harga relatif mahal.
Kapasitas penyimpanan energi listik besar.	<i>Lifecycle</i> baterai Lipo pendek, sekitar 300-400 kali siklus pengisian ulang, namun juga bergantung pada perlakuan yang diberikan pada baterai.
Tingkat <i>discharge rate</i> energi tinggi.	Alasan keamanan, Lipo memakai bahan elektrolit yang mudah terbakar.

## 2.12. Website

Internet merupakan sebuah jaringan komputer yang sudah terkoneksi kedalam jaringan global. Internet menyediakan halaman informasi atau *website* [8]. *Website* merupakan komponen atau kumpulan komponen yang terdiri dari teks, gambar, suara animasi sehingga lebih merupakan media informasi yang menarik untuk dikunjungi. *Website* secara garis besar digolongkan menjadi *website* statis, *website* dinamis dan *website* interaktif. Dalam pembuatan *website* yang sederhana dapat menggunakan *tools* diantaranya HTML (*HyperText Markup Language*) digunakan untuk memudahkan tampilan desain kolom, PHP digunakan untuk menampilkan data, fungsi pengolahan data dan fungsi – fungsi lainnya dan MySQL digunakan untuk mengolah *database*, menyimpan dan mengambil data. Supaya dapat menghasilkan tampilan wujud yang terintegrasi performatan hiperteks sederhana ditulis dalam berkas format ASCII sehingga menjadi sebuah halaman *web*. Pembuatan *website* pada umumnya menggunakan bahasa PHP (*Personal Homepage*) *Tools*. Skrip ini akan membuat suatu aplikasi dapat di integrasikan kedalam HTML sehingga suatu halaman HTML tidak lagi bersifat statis, namun bersifat dinamis. Sifat *server side* ini membuat pengerjaan skrip tersebut dikerjakan di *server*.

### 2.12.1.Database MySQL

MySQL merupakan *software* resmi yang dikembangkan sebuah perusahaan Swedia bernama MySQL AB, sebelumnya bernama TcX Data Konsult AB. Pada awalnya MySQL memakai nama mSQL atau mini SQL. Sebagai antarmuka yang digunakan, ternyata dengan menggunakan mSQL, itu mengalami banyak hambatan yaitu sangat lambat dan tidak fleksibel. Oleh karena itu, Michael Widenius berusaha mengembangkan *interfaces* tersebut hingga ditemukan MySQL. Kala itu MySQL di distribusikan secara khusus, yakni untuk keperluan non-komersial bersifat gratis, sedangkan untuk kebutuhan komersial diharuskan membayar lisensi. Barulah sejak versi 3.32.19, MySQL, dikategorikan *software* berlisensi GPL, yakni dapat dipakai tanpa biaya untuk kebutuhan apapun. Hingga kini MySQL dapat dijalankan diberbagi sistem operasi misalnya, *Linux*, *UNIX*, *Windows*. Kelebihan MySQL adalah pada kecepatan akses, biaya, konfigurasi, tersedia *source code* karena MySQL dibawah *Open Source License* dan menjadi

database open source yang sangat populer. MySQL merupakan *database relational* (RDBMS) yang mempunyai kemampuan yang sangat cepat untuk menjalankan SQL dengan multithread dan multi user. Oleh karena itu dengan melihat begitu tingginya potensi MySQL. Segala fitur pendukung terus dikembangkan agar penggunaan MySQL dapat lebih optimal [8].

Berikut ini beberapa kelebihan MySQL sebagai database server antara lain :

- a. *Source MySQL* dapat diperoleh dengan mudah dan gratis.
- b. Sintaksnya lebih mudah dipahami dan tidak rumit.
- c. Pengaksesan database dapat dilakukan dengan mudah.
- d. MySQL merupakan program yang multithreaded, sehingga dapat di pasang pada server yang memiliki multi PI.
- e. Mendukung program-program umum seperti C, C++, *Java*, *Perl*, *PHP*, *Python*, dsb.
- f. Bekerja pada berbagai platform (tersedia berbagai versi untuk berbagai sistem operasi).
- g. Memiliki jenis kolom yang cukup banyak sehingga memudahkan konfigurasi sistem *database*.
- h. Memiliki sistem keamanan yang cukup baik dengan verifikasi host.
- i. Mendukung ODBC untuk sistem operasi *windows*.
- j. Mendukung record yang memiliki kolom dengan panjang tetap atau panjang bervariasi.

MySQL dan PHP merupakan sistem yang saling terintegrasi. Maksudnya adalah pembuatan database dengan menggunakan sintak PHP dapat dibuat sedangkan input yang dimasukkan melalui aplikasi web yang menggunakan *script server side* seperti PHP dapat dimasukkan ke database MySQL, yang ada di server dan tentunya web tersebut berada di sebuah *web server*.

*Web server* adalah sebuah teknologi informasi yang memproses suatu permintaan melalui HTTP (*HyperText Transfer Protocol*), sebuah protokol jaringan dasar yang digunakan untuk mengirimkan informasi ke *world wide web*. Fungsi utama dari sebuah *web server* adalah untuk menyimpan, memproses dan mengirim sebuah halaman web kepada klien. Komunikasi antara klien dan *server* melalui layanan HTTP. Halaman dikirim dalam bentuk dokumen HTML (*HyperText Markup Language*), yang dapat berisi gambar, lembar kerja dan sebuah catatan yang

ditambahkan

pada teks. *Web server* tidak selalu melayani basis *world wide web*, dapat juga ditemukan pada sebuah sistem terintegrasi seperti *printer, router, webcam* dan layanan jaringan lokal [7].

### 2.12.2. *Hypertext Preprocessor (PHP)*

PHP merupakan singkatan dari "*Hypertext Preprocessor*", PHP adalah sebuah bahasa scripting atau sering disebut bahasa pemrograman yang terpasang pada HTML. Sebagian besar sintaknya mirip dengan bahasa pemrograman C, Java, Asp dan Perl, ditambah beberapa fungsi PHP yang spesifik dan mudah dimengerti. Sejarah PHP, awalnya PHP merupakan kependekan dari Personal Home Page (situs personal) dan PHP itu sendiri pertama kali dibuat oleh Rasmus Lerdorf pada tahun 1995, dan pada saat PHP masih bernama FI (*Form Interpreter*), yang wujudnya berupa sekumpulan Script yang digunakan untuk mengolah data form dari web dan selanjutnya Rasmus merilis kode sumber tersebut untuk umum dan memakannya PHP. Kegunaan dari PHP adalah untuk membuat tampilan web menjadi lebih dinamis, dengan PHP bisa menampilkan atau menjalankan beberapa file dalam 1 file dengan cara di *include* atau *require*, dan PHP itu sendiri sudah bisa berinteraksi dengan beberapa database walaupun dengan kelengkapan yang berbeda, yaitu seperti DBM, *FilePro*, *PostgreSQL*, *MySQL*.

Prinsip kerja website pada umumnya menggunakan bahasa pemrograman PHP. PHP memiliki kinerja yang multiguna dalam pembuatan desain web. Prinsip kerja PHP diawali dengan permintaan suatu halaman web (*file.php*) oleh browser atau klien. Kemudian berdasarkan alamat di internet (URL) *browser* mendapatkan alamat dari web server yang akan mengidentifikasi halaman yang diminta, dan menyampaikan segala informasi yang diminta, dan menyampaikan segala informasi yang dibutuhkan oleh *web server*. Selanjutnya, ketika file PHP yang diminta didapatkan *web server*, isinya segera dikirimkan ke mesin PHP untuk diproses dan memberikan hasilnya (berupa kode HTML) ke *web server*, lalu menyampaikannya ke klien.

### 2.12.3. *Hypertext Markup Language (HTML)*

HTML adalah bahasa dari World Wide Web (WWW) yang dipergunakan untuk menyusun dan membentuk dokumen agar ditampilkan pada program browser. Tiap kali mengakses dokumen web, maka sesungguhnya mengakses dokumen yang ditulis dengan format HTML. Beberapa orang merasa keberatan jika dikatakan HTML, adalah sebuah bahasa pemograman karena struktur yang dimilikinya dianggap terlalu sederhana, kode-kode dibaca oleh *browser* baris per baris dari atas kebawah. HTML juga tidak memiliki 'looping' seperti bahasa pemograman lain. Pada HTML dipergunakan *hyperteks link* atau hubungan antara teks dan dokumen lain. Dengan demikian pembaca dokumen bisa melompat dari satu dokumen ke dokumen yang lain dengan mudah. HTML adalah halaman web biasanya disusun dari kode-kode html yang disimpan dalam sebuah file yang berektensi \*.html file html ini dikirimkan oleh *server* (atau file) ke *browser*, kemudian *browser* menerjemahkan kode kode tersebut sehingga menghasilkan sebuah tampilan yang indah. Lain halnya dengan program php, program ini harus diterjemahkan oleh *webserver* agar dapat ditampilkan.

.....*Halaman ini sengaja dikosongkan*.....



## **BAB III**

### **PERANCANGAN DAN REALISASI ALAT**

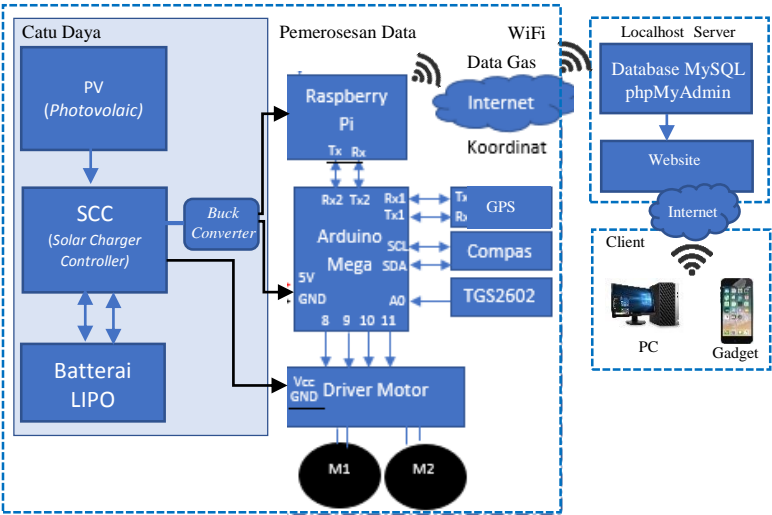
Pada bab ini menjelaskan tentang perancangan alat yang meliputi perancangan sistem, perancangan perangkat keras, perancangan perangkat lunak, dan Perancangan *website*.

#### **3.1. Perancangan Sistem**

Mobile robot bertenaga surya untuk pemetaan konsentrasi gas tergabung dalam sistem besar, terdiri dari *Power Station*, *Wireless Charging*, *Autodocking* Pengisian Daya, *Obstacle Detector* dan Pemetaan Konsentrasi Gas. Untuk mendukung sistem dari mobile robot ini, diperlukannya *power station* yang merupakan tempat untuk pengisian daya apabila daya baterai pada mobile robot habis selama beroperasi. Metode pengisian (*charging*) daya pada mobile robot di *Power Station* menggunakan *Wireless Charging*, dengan hal tersebut mobile robot tidak memerlukan kabel dalam pengisian daya pada *Power Station* saat *charging* baterai. Penempatan jarak antara mobile robot dengan *Power Station* dalam pengisian daya secara *wireless* ditentukan secara otomatis dengan sistem *Autodocking* pengisian daya, dimana robot akan secara otomatis mencari *Power Station* terdekat apabila mobile robot akan kehabisan daya baterai. Untuk impelentasinya, mobile robot akan beroperasi pada tempat eksploitasi minyak, sehingga akan memungkinkan objek-objek yang dapat menghalangi lintasan (*waypoint*) mobile robot dapat dihindari dengan sistem *Obstacle Detector*. Pada dasarnya mobile robot akan terus berkerja dalam pemetaan konstansi gas selama 24 jam, sehingga membutuhkan sumber daya tambahan berupa panel surya (PV) dalam membantu mobile robot beroperasi dengan waktu yang relatif lama. Terlepas dari sistem besar tersebut, pada tugas akhir ini akan membahas mobile robot dalam pemetaan konsentrasi gas dengan sumber daya tambahan PV.

Dalam sistem kejanya, pada penelitian ini *Mobile Robot* akan ditentukan oleh pengguna titik-titik *waypoint* yang akan dilaluinya. Penentuan *waypoint* berdasarkan nilai latitude dan longitude dari GPS. *Waypoint* tersebut ditentukan secara manual berupa titik koordinat suatu tempat yang dipilih untuk pemetaan konsentrasi gas . Robot akan bergerak secara otomatis mengikuti *waypoint* yang telah ditentukan dengan menggunakan metode

navigasi *heading* dari kompas. Selama perjalanan, apabila mobile robot mensensing konsentrasi gas maka data dari konsentrasi gas pada wilayah yang dilaluinya tersebut akan dikirim ke internet kemudian ditampilkan dalam bentuk web yang sudah terintegrasi dengan Google *map*. Blok diagram sistem dapat ditunjukkan pada Gambar 3.1 dan Spesifikasi dari Mobile Robot Bertenaga Surya Untuk Pemetaan Konsentrasi Gas pada tugas akhir ini dapat dilihat pada Tabel 3.1



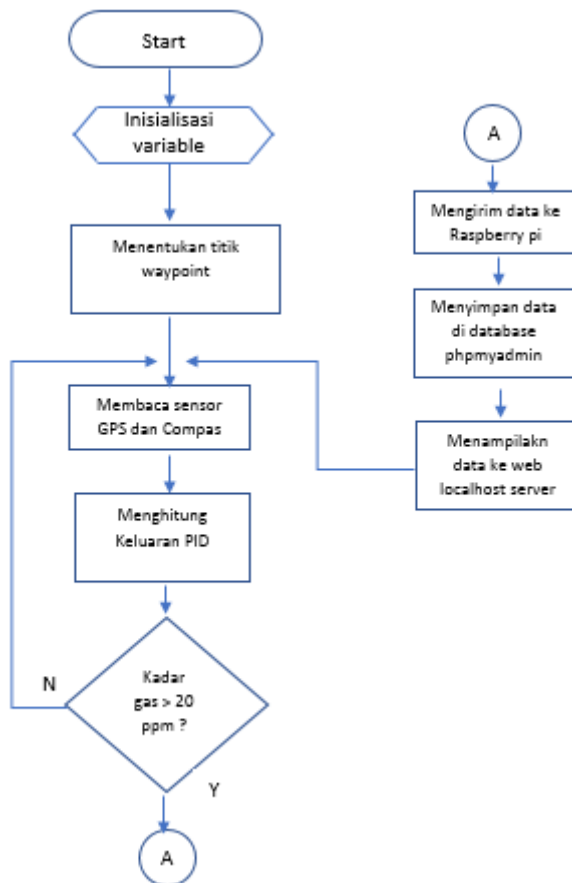
Gambar 3.1 Blok diagram sistem

Tabel 3.1. Spesifikasi *Mobile Robot*

Dimensi	40 x 35 x 23 cm
Berat Total	5 Kg
Jumlah Motor	4 Buah
Jumlah Panel Surya	1 Modul 20Wp
Mode Kendali	<i>Auto waypoint</i>
Monitoring Data	<i>Web Hosting</i>

Dari Gambar 3.1, secara umum sistem menggunakan Arduino Mega 2560 sebagai pengolah data. Suplai daya yang dibutuhkan sistem bersumber dari modul panel surya yang dimana daya disimpan pada baterai Lipo melalui sebuah PWM *Solar Charge Controller*. Ketika batterai mendekati penuh, SCC dengan sistem PWM *step by step* menurunkan jumlah daya yang masuk ke baterai untuk mengurangi kejenuhan pada baterai tersebut. Saat voltase di baterai dalam keadaan hampir kosong, maka controller berfungsi menghentikan pengambilan arus listrik dari baterai oleh beban. Dalam kondisi voltase tertentu (umumnya sekitar 10% sisa voltase di baterai), maka pemutusan arus beban dilakukan oleh *controller*. Dengan regulasi diatas, proses pengisian daya pada Lipo akan menjadi aman.

Seperti yang dijelaskan sebelumnya. Ketika robot berjalan, data watpoint yang telah ditentukan secara urut akan menjadi titik-titik tujuan yang harus dilaluinya. Sensor GPS secara periodik memberikan informasi data lokasi robot berada, dengan bantuan sensor kompas yang memberikan arah robot. Informasi data tersebut masuk ke dalam pemroses data yaitu Arduino Mega2560, selanjutnya data tersebut digunakan sebagai acuan. Koordinat lokasi sebagai menjadi data jarak dan arah yang harus dituju antara *waypoint* dan robot, data kompas menjadi arah *heading* robot ketika meunju *waypoint*. Data kompas juga meberikan nilai selisih sudut yang digunakan kendali untuk mengatur PWM motor, agar robot bisa bermanuver ke arah sudut yang harus dituju. Selama robot berjalan dan mendeteksi kadar konsentrasi gas berbahaya yaitu lebih dari 20 ppm yang dibaca oleh sensor gas, data dan lokasi konsentrasi gas tersebut akan dikirim ke mini PC Raspberry Pi melalui serial. Data yang diterima oleh Raspberry akan disimpan di Database kemudian ditampilkan ke web server dengan perantara internet . Dengan hal tersebut, *client* dapat melihat data hasil pemetaan konsentrasi gas lewat *browser* melauai *gadget* yang terhubung dengan internet. Diagram alir sistem dapat dilihat pada Gambar 3.2.

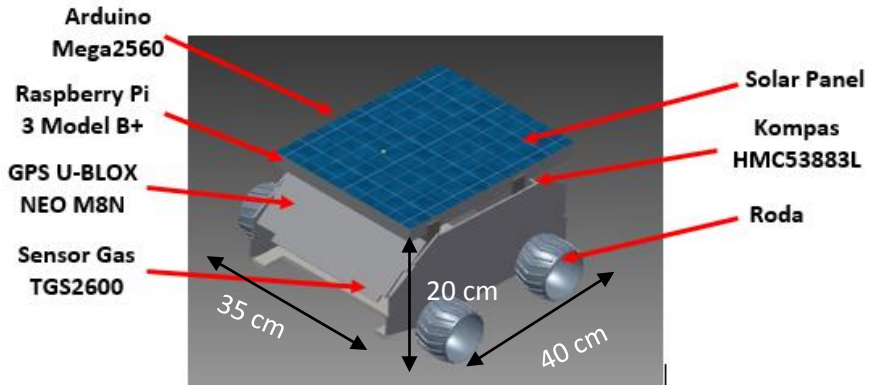


Gambar 3.2 Flowchart Sistem

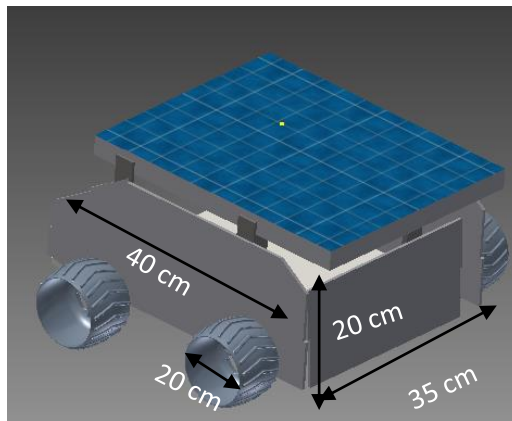
### 3.2. Perancangan Perangkat Keras (LAMPIRAN A)

Perancangan perangkat keras sistem kendali yaitu penggabungan fungsi beberapa komponen seperti panel surya, sensor-sensor, mikrokontroller, driver motor, antena dan aktuator. Semua komponen dirangkai demikian rupa seperti yang ditunjukkan pada Gambar 3.1. Model robot yang digunakan adalah *mobile* robot dengan penggerak motor DC 4 buah (4WD) yang masing-masing menggunakan satu roda pada tiap motor seperti

yang ditunjukkan pada Gambar 3.3. Sedangkan sistem skematik rangkaian mobile robot dapat ditunjukkan pada Gambar 3.4.

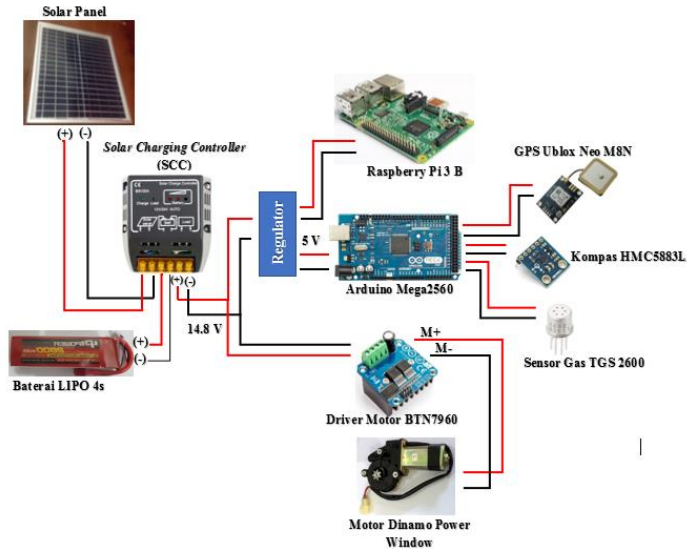


(a)



(b)

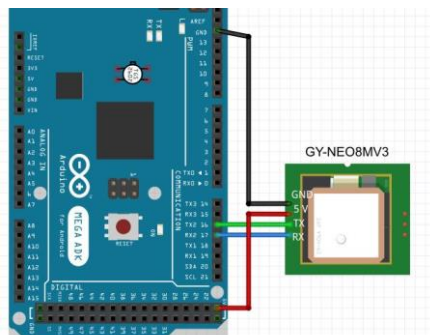
Gambar 3.3 (a) Desain robot Autodesk Inventor tampilan depan  
(b) Desain robot Autodesk Inventor tampilan belakang



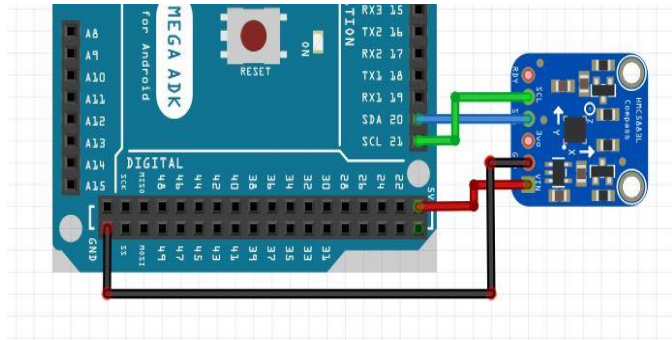
Gambar 3.4 Blok diagram sistem elektronis

### 3.2.1. Perancangan Sensor GPS Ublox Neo M8

GPS Ublox Neo M8 menggunakan komunikasi serial USART TTL yang dihubungkan pada pin Arduino. Dengan model *cross connection* dimana Rx GPS terhubung Tx Arduino, dan Tx GPS terhubung Rx Arduino. Skema rangkaian jalur konfigurasi rangkaian GPS Ublox Neo M8 dengan Arduino Mega dapat dilihat pada Gambar 3.5.



Gambar 3.5 Schematic GPS dengan Arduino Mega2560



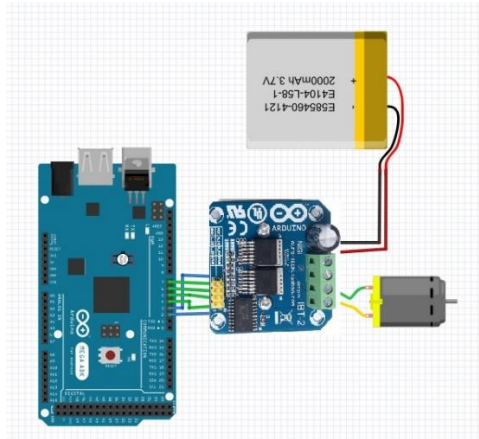
Gambar 3.6 Schematic kompas dengan Arduino Mega2560

### 3.2.2. Perancangan Sensor Kompas

Sensor kompas digunakan sebagai arah *heading* robot. Sensor ini menggunakan komunikasi *inter-integrated communication* (I2C). Keunggulan dari komunikasi I2C ini adalah dapat menanamkan banyak perangkat hanya dengan menggunakan dua buah kabel komunikasi. Kelemahannya adalah kecepatan transfer data yang cukup lambat, mulai dari 100 kHz sampai 400 kHz. Konfigurasi rangkaian sensor kompas HMC5883L dengan Arduino Mega ditunjukkan pada Gambar 3.6.

### 3.2.3. Perancangan Rangkaian Driver Motor BTN7960

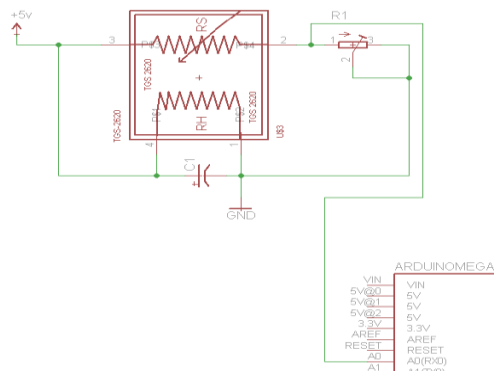
Modul driver motor BTN7960 merupakan modul H-bridge yang berbasis dua IC BTN7960 *Half-Bridge* yang dikombinasikan menjadi satu. Konfigurasi dari rangkaian *driver* motor dengan Arduino ditunjukkan pada Gambar 3.7.



Gambar 3.7 Rangkaian *driver* motor dengan Arduino

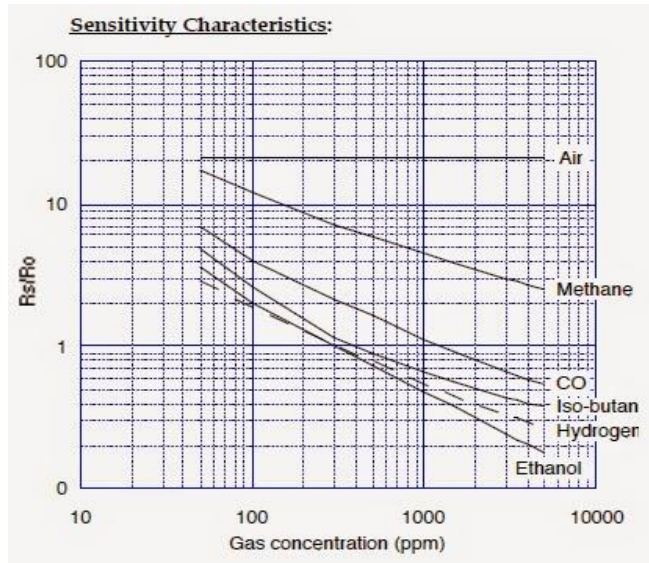
### 3.2.4. Perancangan Sensor Gas TGS2602

Sensor Gas TGS2602 mempunyai hasil keluaran berupa tegangan jika mendeteksi kadar gas di udara. Pin keluaran sensor gas ini dihubungkan dengan Pin A0 pada Arduino. Sensor ini disupply dengan tegangan 5 volt. Pada Gambar 3.8, pin 1 dan pin 4 yang merupakan sumber daya untuk resistor pemanas atau  $R_h$ , dihubungkan paralel dengan sumber daya  $V_c$ . Ini diperlukan karena rangkaian pemanas digunakan untuk memperbaiki elemen sensor pada temperatur tertentu untuk penginderaan secara optimal.



Gambar 3.8 Rangkaian skematik sensor gas dengan Arduino Mega2560

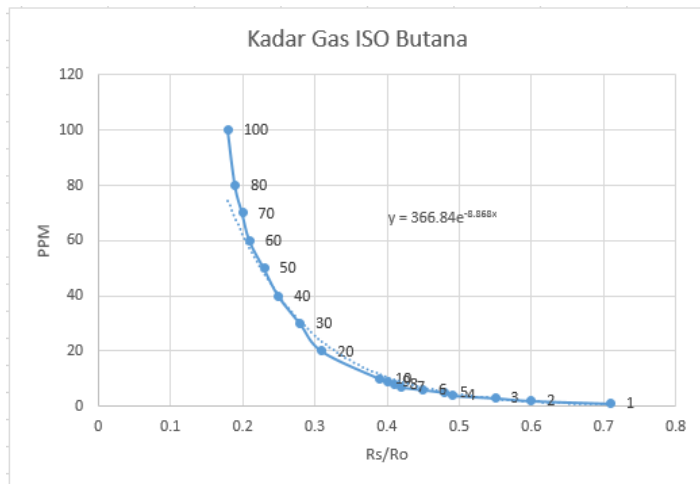




Gambar 3.9 Grafik perbandingan  $R_o/R_s$  dengan kadar gas[2]

Tabel 3.2. Perbandingan  $R_s/R_o$  dengan kadar gas

No	$R_o/R_s$	Ppm
1	0.71	10
2	0.6	20
3	0.55	30
4	0.49	40
5	0.48	50
6	0.45	60
7	0.42	70
8	0.41	80
9	0.4	90
10	0.39	100



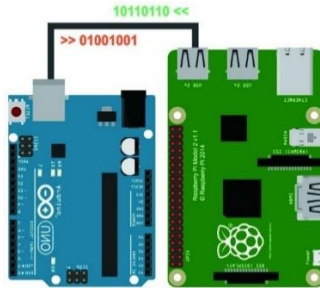
Gambar 3.10 Grafik persamaan ppm gas iso-butana

Untuk mendapatkan nilai kadar gas, merujuk pada grafik sensor gas yang digunakan. Dalam penelitian ini menggunakan data grafik sensor TGS2600. Nilai dari grafik pada Gambar 3.10 kemudian diambil dan dimasukkan pada tabel. Sehingga mendapatkan tabel perbandingan antara  $R_o/R_s$  dengan ppm gas seperti yang ditunjukkan pada Tabel 3.2.

### 3.2.5. Koneksi Raspberry Pi

*Raspberry pi* yang digunakan adalah *rasberry pi 3 model B* dengan menggunakan sistem operasi *raspbian versi stretch*. Koneksi yang digunakan pada *arduino* dengan *raspberry pi* menggunakan serial komunikasi melalui *universal serial bus* (USB) terlihat pada gambar 3.12.

*Rapsberry pi* membutuhkan beberapa aplikasi dalam membangun sebuah server. Pada tugas akhir ini, aplikasi yang ter-install diantaranya *web server apache2*, *php7*, *database MySQL* dan *phpmyadmin*. *Apache* merupakan aktivasi web lokal server dengan alamat IP 127.0.0.1 atau yang sering disebut *localhost*. *PHP* sebagai pembangun *web server* dinamis yang terhubung dengan *database server*. *MySQL* merupakan *database server* untuk menyimpan data yang dikirim dari *arduino*. *PhpMyAdmin* sebagai konfigurasi dari *database MySQL*.



Gambar 3.11 Antarmuka Raspberry Pi dengan Arduino

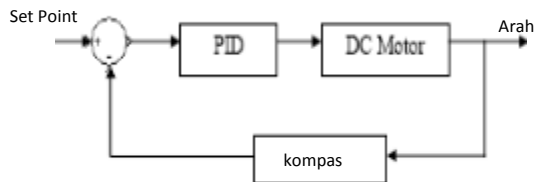
### 3.3. Perancangan Perangkat Lunak (LAMPIRAN B)

Perancangan perangkat lunak ini digunakan untuk mengintegrasikan dan memfungsikan modul-modul perangkat keras dengan pengguna. Perancangan ini meliputi desain kendali PID, penunjukan arah dari GPS, mendapatkan garis lintang dan garis bujur dari GPS, dan sudut *waypoint* yang dituju.

#### 3.3.1. Desain Kendali PID Pada *Differential Speed Steering*

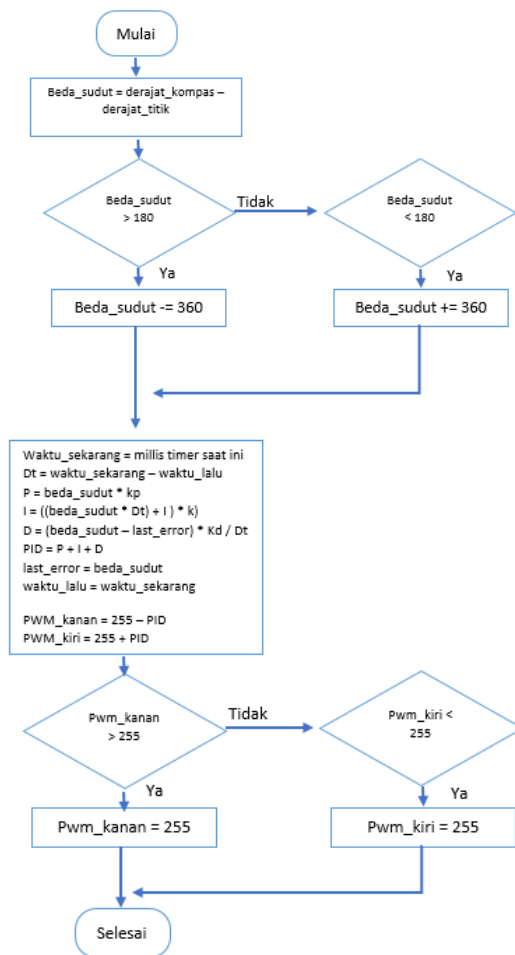
Desain kendali ini digunakan robot dalam bermanuver mengikuti sudut acuan dari hasil pengolahan data koordinat yang ditentukan pada robot. Diagram blok pada kendali ini ditunjukkan pada Gambar 3.13. Nilai dari *setpoint* didapatkan sudut Mobile Robot dengan titik koordinat yang dituju diakses dari GPS dengan kode program seperti berikut:

```
jarak_titik = (unsigned
long)TinyGPSPlus::distanceBetween(gps.location.lat(),
gps.location.lng(),titik_lat, titik_lon);
derajat_titik = TinyGPSPlus::courseTo(gps.location.lat(),
gps.location.lng(), titik_lat, titik_lon);
```



Gambar 3.12 Diagram blok kendali PID [2]

Sedangkan nilai  $K_p$ ,  $K_d$ , dan  $K_i$  diberikan sesuai dengan *trial and error* respon sistem yaitu :  $K_p = 5.0$ ,  $K_d = 7.2$  dan  $K_i = 0.000035$ . Setelah sudut mobile robot didapatkan dan nilai parameter PID sudah ditentukan, kemudian menyusun program algoritma agar robot dapat bermanuver ke arah sudut tersebut. Program alir dari algoritma disusun dalam *flowchart* sebagai berikut :



Gambar 3.13 Diagram alir algoritma PID

Algoritma yang telah disusun menjadi diagram blok, kemudian diterjemahkan ke dalam kode program. Kode program dari daigram alir Gambar 3.13 adalah sebagai berikut:

```
beda_sudut = derajat_kompas - derajat_titik;
if(beda_sudut < -180){
    beda_sudut += 360;
}
else if(beda_sudut > 180){
    beda_sudut -= 360;
}

waktu_sekarang = millis();
dt = (float)(waktu_sekarang - waktu_lalu);

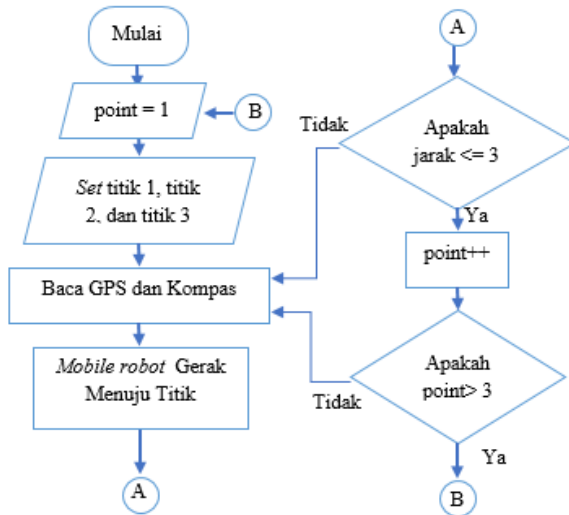
float P = beda_sudut * kp;
float I = ((beda_sudut*dt) + I) * ki;
float D = ((beda_sudut - last_error) * kd) / dt;

PID = P + I + D;

last_error = beda_sudut;
waktu_lalu = waktu_sekarang;

pwm_kanan = (255 - PID);
pwm_kiri = (255 + PID);

if(pwm_kanan > 255){
    pwm_kanan = 255;
    if(pwm_kiri < 0){
        belok_kiri();
    }
    else{
        maju();
    }
}
if(pwm_kiri > 255){
    pwm_kiri = 255;
    if(pwm_kanan < 0){
        belok_kanan();
    }
    else{
        maju();
    }
}
```



Gambar 3.14 Diagram alir algoritma *waypoint*

### 3.3.2. Navigasi *Waypoint*

Pada tugas akhir ini, terdapat tiga titik koordinat yang dimasukkan kedalam program seperti yang ditunjukkan pada Gambar . GPS dan kompas mengatur navigasi untuk melalui titik-titik koordinat tersebut. Sudut Mobile robot terhadap titik yang dituju memiliki rentang nilai dari 0 hingga 359 derajat. Pertambahan nilai sudut tersebut sesuai arah jarum jam.

Sudut Mobile robot didapat dari nilai kompas. Dimana nilai sudut kompas juga memiliki rentang nilai dari 0 hingga 359 derajat. Metode untuk menggerakkan Mobile robot secara otomatis menuju titik yang dituju yaitu dengan membandingkan antara sudut yang didapat dari kompas dengan sudut dari GPS, serta dengan memanfaatkan nilai jarak ke titik tujuan.

Selanjutnya, ketika sudah berjarak 3 meter (jarak dapat disesuaikan, pada tugas akhir ini diatur berjarak 3 meter) dari titik tujuan pertama, maka titik tujuan berubah menjadi titik tujuan kedua. Begitu pula hingga titik ketiga. Ketika sudah lebih dari titik ketiga, maka titik kembali ke titik 1 kembali. Hal ini pada program ditulis sebagai berikut:

```

point = 1;
switch(point){
    case 1: titik_lat = -7.284806;
            titik_lon = 112.796691;
            break;
    case 2: titik_lat = -7.284735;
            titik_lon = 112.796745;
            break;
    case 3: titik_lat = -7.284734;
            titik_lon = 112.796836;
            break;
}

if(jarak_titik <= 3){
    point++;
}

if(point > 3){
    point = 1;
}

```

### 3.3.3. Mengakses Modul GPS Ublox Neo M8N

Untuk mengakses data GPS, baudrate antara modul GPS dengan penerima harus sama. Pada tugas akhir ini baudrate GPS menggunakan baudrate 9600. Untuk mendapatkan nilai latitude dan longitude dari modul GPS, pada tugas akhir ini menggunakan *library* TinyGPS++. Kode dasar untuk membaca data masukan dari jalur Serial1.

```

#include <TinyGPS++.h>
TinyGPSPlus gps;
static const uint32_t GPSPBaud = 9600;

void setup() {
    Serial.begin(9600);
    Serial1.begin(GPSPBaud);
}

Void loop(){
    float altitude = gps.location.lat();
    float longitude = gps.location.lng();
}

```

```

jarak_titik = (unsigned
    long)TinyGPSPlus::distanceBetween(gps.location.lat(),
    gps.location.lng(),titik_lat, titik_lon);
}

```

Dari program di atas, nilai latitude dan longitude dari modul GPS uBlox Neo M8 dapat diakses dengan menggunakan fungsi `gps.location.lat()` dan `gps.location.lng()` dari library TinyGPS++. Untuk fungsi `distantanceBetween()` digunakan untuk mengetahui jarak antara titik navigasi yang dituju dengan titik latitude dan longitude yang sekarang. Hasil dari perhitungan fungsi tersebut sudah dikonversi dalam satuan meter (m).

### 3.3.4. Mengakses Modul Kompas HMC53883L

Modul kompas HMC5388L menggunakan komunikasi I2C (*Inter Integrated Circuit*) sebagai protokolnya. Sehingga untuk membaca modul ini, harus mengetahui alamat dari setiap register yang akan diakses. Alamat register HMC53883L 0x1E, alamat register tulis 0x3C, alamat register data 0x03. Prosedur pembacaan sensor kompas HMC53883L adalah sebagai berikut :

```

Void kompas(){
Wire.beginTransmission(0x1E);
Wire.write(0x03);
Wire.requestFrom(0x1E,6);
    if(Wire.available() <= 6)
    {
        for(int i=0;i<6;i++)
        {
            outputData[i]=Wire.I2C_RX();
        }
    }
x=outputData[0] << 8 | outputData[1];
z=outputData[2] << 8 | outputData[3];
y=outputData[4] << 8 | outputData[5];
angle = (double)atan2(y, x);
angle += declinationAngle;

if (angle < 0)  angle += 2*PI;
if (angle > 2*PI) angle -= 2*PI;
float bearing = angle * 180/PI; }

```



Data dari masing-masing sumbu mempunyai lebar data 16 bit. Pembacaan dibagi menjadi dua, yaitu pembacaan MSB sepanjang 8 bit pertama, kemudian LSB pada 8 bit kedua. Data dari sumbu x dan sumbu y kemudian dimasukkan dalam fungsi  $\text{atan2}(y, x)$  untuk menghasilkan *arc tangen* dalam empat kuadran pada sumbu x dan y tertentu. Sumbu z tidak digunakan dalam formula, karena kompas dianggap selalu dalam keadaan tegak terhadap garis normal. Hasil formula tersebut menghasilkan nilai sudut dengan satuan radian bernilai antara  $-3.14/2$  hingga  $3.14/2$ . Untuk merubahnya ke bentuk derajat hasil dari formula tersebut dikalikan dengan  $180/\text{PI}$ .

### 3.3.5. Mengakses Sensor Gas TGS2600

Dari hasil plotting yang menghasilkan persamaan pada Gambar, maka program untuk pembacaan gas adalah sebagai berikut:

```
Void pembacaan_kadarGas(){
    float pembacaan = analogRead(A0);
    float const eksponen = 2.718282;
    float batas = awal_ADC;
    float level = map(pembacaan, batas, 1023, 0, 100);
    float tegangan_adc = pembacaan * (5.0 / 1023.0);
    float rs = (4.9 * 10000.0) / tegangan_adc;
    rs -= 10000.0;
    float perbandingan = rs / ro;
    float nilai = (-8.868) * perbandingan;
    float nilai_eksponen = pow(eksponen, nilai);
    kadar_gas = 366.84 * nilai_eksponen;
    if(kadar_gas < 0 ){
        kadar_gas = 0;
    }
}
```

### 3.3.6. Mengakses Driver Motor

Untuk mengakses perancangan modul *driver* motor BTN7960 pada Arduino seperti yang ditunjukkan pada Gambar, secara sederhana programnya dapat ditulis sebagai berikut :

```
void motor(){  
  analogWrite(2, pwm_kanan);  
  analogWrite(3, pwm_kanan);  
  
  analogWrite(4, pwm_kiri);  
  analogWrite(5, pwm_kiri);  
}
```

### 3.4. Perancangan Catu Daya

Perancangan catu daya *Mobile* Robot digunakan untuk mengetahui berapa daya minimum, torsi minimum dan kapasitas baterai untuk menggerakkan *Mobile* Robot di lapangan. Dengan perancangan tersebut, didapatkan berapa daya panel surya dan kapasitas baterai yang akan dibutuhkan untuk menjalankan *Mobile* Robot.

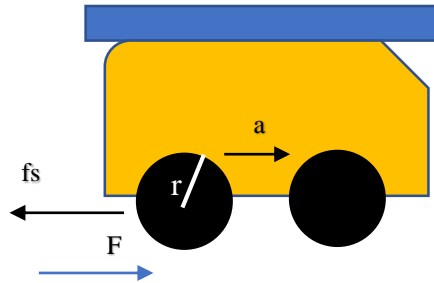
#### 3.4.1. Perhitungan Torsi

Perhitungan torsi digunakan untuk mengetahui berapa kuat gaya dorong untuk mendorong berat dari *Mobile* Robot oleh motor DC yang digunakan. Dalam perhitungan torsi tersebut mengacu pada ilustrasi Gambar 3.16. Gaya yang dapat mendorong *Mobile* Robot dapat dihitung dengan menggunakan Hukum Newton II yang dimana dapat ditulis dengan persamaan (3.1).

$$F = m \cdot a \quad (3.1)$$

Keterangan :

F        = Gaya (N)  
m        = Massa (kg)  
a        = Percepatan (m/s<sup>2</sup>)



Gambar 3.15 Ilustrasi perhitungan torsi

Apabila gaya gesek ( $f_s$ ) diabaikan, gaya yang berkerja pada *Mobile Robot* dengan berat robot mengacu data Tabel 3.1 dan percepatan terhadap *Mobile Robot*  $5 \text{ m/s}^2$  dengan persamaan 3.1 yaitu 25 N. Dari gaya yang berkerja dapat diketahui torsi roda yang dapat mendorong *Mobile Robot* dengan persamaan (3.2)

$$T_{\text{roda}} = r \cdot F \quad (3.2)$$

Keterangan :

- $T_{\text{roda}}$  = Torsi Roda (N.m)
- $F$  = Gaya (N)
- $r$  = Jari-jari roda (m).

Mengacu pada jari-jari roda yang digunakan terukur 10 cm. Sehingga torsi roda untuk mendorong *Mobile Robot* dengan persamaan 3.2 yaitu 2.5 N.m. Sedangkan torsi motor DC yang dapat mendorong massa dari *Mobile Robot* yaitu 125% dari torsi roda sehingga torsi total motor DC yaitu 3.12 N.m. Karena *Mobile Robot* memiliki 4 motor DC sebagai penggerak, maka masing-masing torsi yang ada di motor DC yaitu 0.78 N.m. Hasil perhitungan torsi yang didapat merupakan torsi minimum dari motor DC dalam mendorong massa *Mobile Robot* dengan percepatan konstant yaitu  $5 \text{ m/s}^2$ .

### 3.4.2. Perhitungan Daya

Perhitungan pada perancangan ini digunakan untuk mengetahui berapa daya panel surya yang akan digunakan untuk *Mobile Robot*. Dari hasil perhitungan torsi minimum motor DC dapat diketahui berapa daya yang dibutuhkan untuk menghasilkan torsi tersebut dengan menggunakan persamaan 3.3

$$P = T \cdot \omega \quad (3.3)$$

Keterangan :

P = Daya (Watt)  
 T = Torsi Motor DC (N.m)  
 $\omega$  = Kecepatan Rotasi (rad/s)

Apabila kecepatan motor DC sesuai dengan *datasheet* diatur pada kecepatan minimum konstant 55 rpm (5.75 rad/s) dalam menggerakkan *Mobile Robot*, dengan menggunakan persamaan 3.3 didapatkan daya yang dihasilkan yaitu 17.94 Watt. Hasil perhitungan daya perancangan merupakan daya minimum yang digunakan dalam menggerakkan *Mobile Robot* dengan rugi-rugi daya diabaikan. Dari hasil tersebut pada Tugas Akhir ini digunakan panel surya dengan daya yang lebih besar dari daya perhitungan yaitu 20 Watt.

### 3.4.3. Perhitungan Daya Tahan Baterai

Estimasi daya tahan baterai merujuk kepada hasil konsumsi arus pada *Mobile Robot* sesuai dengan *datasheet* dari masing-masing komponen. Dari perhitungan total yang diperoleh, arus maksimum *Mobile Robot* saat berkerja yaitu 3200 mA. Dengan demikian, daya tahan baterai dengan menggunakan baterai kapasitas 2600mAh dapat dihitung dengan persamaan (3.4)

$$\text{Kapasitas (mAh)} = \frac{t_{on} (\text{menit}) \times \text{Arus Maksimum (mA)}}{60 (\text{menit})} \quad (3.4)$$

$$\text{Ton (menit)} = \frac{2600 (\text{mAh}) \times 60 (\text{menit})}{3200 (\text{mA})}$$

$$\text{Ton (menit)} = 48.75 \text{ menit}$$

Dari perhitungan daya tahan baterai, *Mobile Robot* dapat beroperasi dengan menggunakan baterai berkapasitas 2600 mAh yaitu selama 48.75 menit. Kapasitas baterai tersebut dipilih karena harganya lebih murah dan beratnya lebih ringan.

### 3.5. Perancangan Website

Perancangan Web dimulai dari penerimaan data dari Arduino ke Raspberry Pi dengan Serial. Data yang sudah diterima akan tersimpan pada database *PhpMyAdmin* kemudian ditampilkan di *Website*. Perancangan dimulai dari konfigurasi IP Address Raspberry, pembuatan database *PhpMyAdmin* dan desain web.

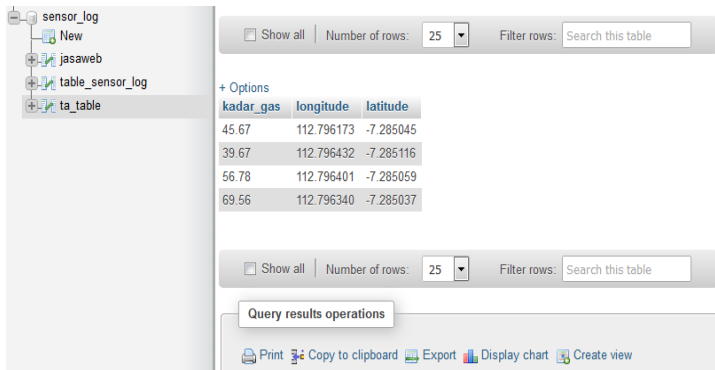
### 3.5.1. Konfigurasi Ip Address Raspberry Pi

Konfigurasi IP Address digunakan untuk mengakses remote *console/terminal* untuk remote *connection* komputer melalui Port SSH atau sebagainya (contohnya *Software Putty*). Untuk melakukan *connection* SSH biasanya IP Address yang digunakan alamat IP *ethernet* pada Raspberry Pi. Untuk mengetahui alamat tersebut, pada Lx Terminal Raspberry Pi dapat diketahui dengan perintah “*ifconfig*”. Dari perintah tersebut, dapat dilihat Ip Address *ethernet* dan *wlan0*. Pada konfigurasi alamat IP *wlan0* akan digunakan sebagai alamat web *localhost* untuk menampilkan website yang dibuat. Tetapi konfigurasi alamat IP *wlan0* tidak *static* (berubah-ubah) tergantung alamat Wifi yang terkoneksi dengan Raspberry Pi. Agar IP *address static* dengan alamat IP yang diinginkan, dapat dilakukan pengaturan konfigurasi “*cat/networks/interface*” pada Lx Terminal.

### 3.5.2. Pembuatan Database MySQL PhpMyAdmin (LAMPIRAN C)

Phpmyadmin merupakan tampilan visual yang digunakan untuk memudahkan user dalam membuat dan memodifikasi database MySQL, baik berupa membuat, menghapus, menambah tabel dan banyak sekali intruksi-intruksi yang dapat dijalankan di program ini. Sebelum menggunakan layanan database, terlebih dahulu diinstal perangkat lunak yang mendukung sistem bekerja sinkron dengan layanan database diantaranya adalah install apache, perangkat lunak *phpMyAdmin* dan membuat akun di *phpMyadmin*, selanjutnya ialah install perangkat lunak untuk akses database phpMySQL.

Cara pembuatan database MySQL pada *localhost server*, hal pertama yang perlu dilakukan yaitu mengakses phpmyadmin pada halaman <http://localhost/phpmyadmin> atau <http://ipaddressraspberry/phpmyadmin> kemudian login sesuai dengan akun yang sudah dibuat sebelumnya. Selanjutnya pembuatan database yang disesuaikan dengan kebutuhan. Pada tugas akhir ini, dibuat database bernama *sensor\_log* dengan tabel *ta\_table* menggunakan data kadar\_gas, longitude dan latitude. Gambar 3.17 merupakan database MySQL phpmyadmin yang telah berhasil dibuat.



Gambar 3.17 Database MySQL *Mobile Robot*

### 3.5.3. Desain *Website* (LAMPIRAN D)

Pada tugas akhir ini, pembuatan *website* dalam monitoring kadar gas berbasis PHP dan HTML dimana skrip yang dapat dijalankan di server. Jadi konsepnya berbeda dengan javascript yang dilakukan pada sisi klien, selain itu PHP bisa digunakan untuk mengakses berbagai macam database seperti Access, Oracle, Mysql dan lain-lain.

Fitur desain *website* yang telah dibuat juga didukung dengan adanya google map, dimana memiliki fungsi untuk menampilkan data koordinat dalam peta kadar gas yang dibaca oleh mobile robot. Proses *upload* data pada website tergantung data yang langsung diterima oleh mobile robot kemudian disimpan pada database dengan *refresh* secara otomatis.

Untuk menjalankan *website* yang telah dibuat, file desain *website index.php*, *header.php*, *footer.php*, *peta.php*, *koneksi.php* dan *ambildata.php* disimpan pada *localhost server* yaitu menggunakan *apache* dengan folder sasaran */var/www/html/www.mobilerobot.com*. Desain website yang telah dibuat terdapat beberapa tab bagian diantaranya: *home*, *peta* dan *about*. Seperti yang ditunjukkan pada gambar berikut.

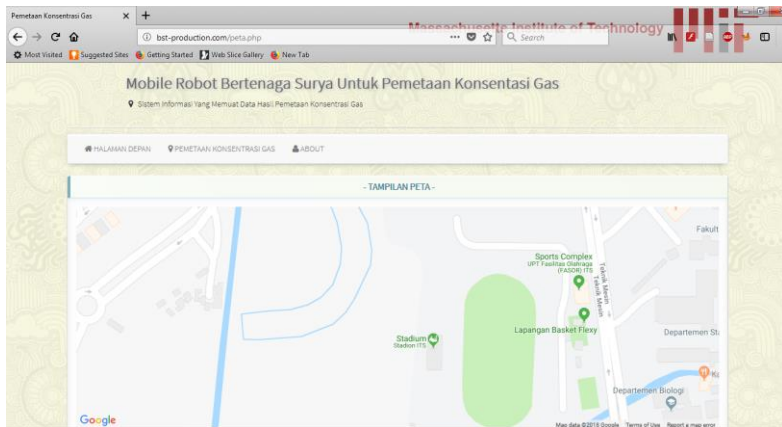


(a)



(b)

Gambar 3.18 a) Tampilan web menu home (PC)  
b) Tampilan web menu home (*Smartphone*)



Gambar 3.19 Tampilan web menu peta (PC)

Gambar 3.18 merupakan tampilan menu home pada website, tampilan ini merupakan tampilan yang pertama kali muncul ketika user mengunjungi website. Sedangkan pada Gambar 3.19 merupakan tampilan menu peta dimana pada tampilan ini akan menampilkan hasil pemetaan kadar konsentrasi gas oleh *Mobile Robot*. Tampilan tersebut dapat diakses oleh *user* dengan cara klik tab menu pada website. Desain website *Mobile Robot* dirancang dapat diakses oleh semua *gadget* dan komputer. Pada tampilan 3.18(b) merupakan tampilan website yang diakses dengan menggunakan *smartphone* iPhone 4s.



## BAB IV

### ANALISA DAN PENGUJIAN SISTEM

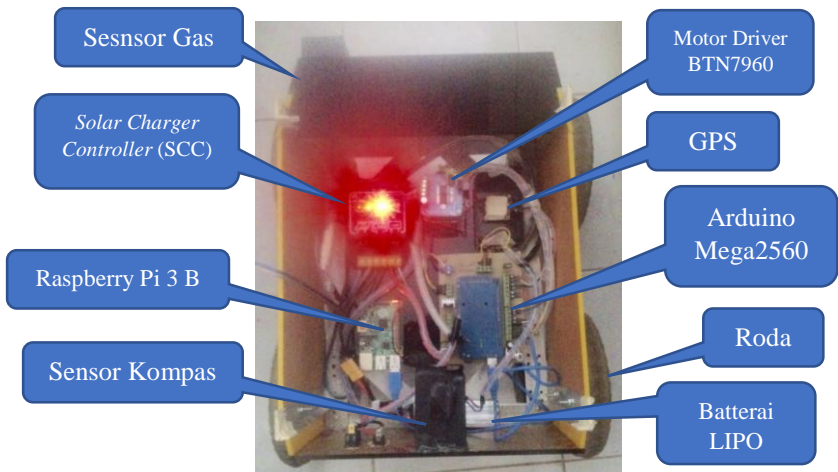
Bab ini menjelaskan tentang pengujian dari sistem yang telah dirancang beserta analisa data yang diperoleh saat pengujian. Pengujian-pengujian yang dilakukan meliputi pengujian sensor kompas, pengujian GPS, pengujian navigasi *waypoint*, pengujian daya yang dihasilkan panel surya, pengujian ketahanan baterai, dan pengujian *uploading* data pada *website*. *Mobile Robot* yang berhasil direalisasikan ditunjukkan pada Gambar 4.1.



(a)



(b)



(c)

Gambar 4.1 (a) Realisasi *Mobile Robot* tampilan depan  
(b) Tampilan belakang (c) Tampilan atas tanpa PV



Gambar 4.2 Lokasi pengujian pada Google Map

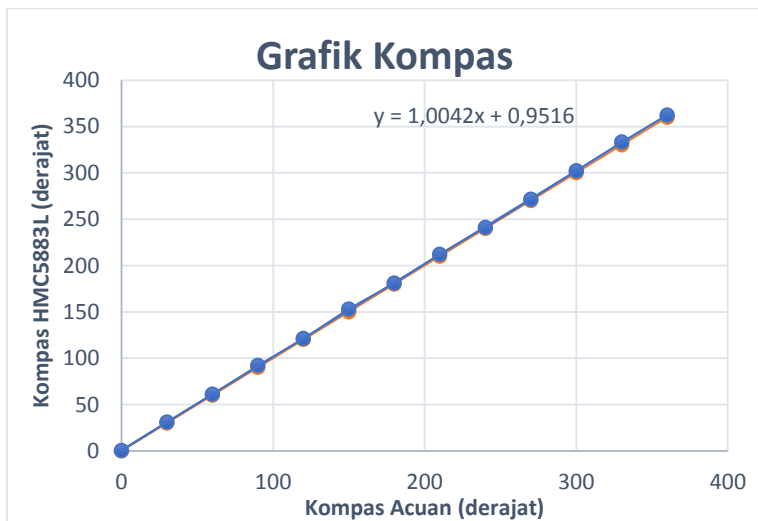
Pengujian mobile robot dilakukan di Departemen Teknik Elektro, Fakultas Teknologi Elektro (FTE), Institut Teknologi Sepuluh Nopember (ITS) , Surabaya. Tempat ini berlokasi di jalan Teknik Mesin, Keputih, Sukolilo, Kota Surabaya, Jawa Timur – 60111. Lokasi tersebut apabila dilihat di Google Maps dapat disajikan pada Gambar 4.2

#### 4.1. Pengujian Sensor Kompas

Pengujian sensor kompas HMC5883L dilakukan dengan cara membandingkan hasil pembacaan sensor dengan sensor kompas acuan. Pada pengujian ini digunakan sensor kompas acuan yaitu sensor kompas di smartphone Iphone Apple 4s. Pembacaan masing-masing sensor dipasang dengan sudut *heading* 0° dan diputar searah jarum jam. Hasil pengujian sensor kompas dapat dilihat pada Tabel 4.1.

Tabel 4.1 Pengujian sensor kompas HMC4883L

NO	Kompas Acuan	Kompas HMC5883L	Error(%)
1	0	0.2	2.00
2	30	31.1	3.54
3	60	61.2	1.96
4	90	92.1	2.28
5	120	121.2	0.99
6	150	153.1	2.02
7	180	179.1	0.50
8	210	212.2	1.04
9	240	241.1	0.46
10	270	271.3	0.48
11	300	302.1	0.70
12	330	333.1	0.93
13	360	359.2	0.19



Gambar 4.3 Grafik pengujian sensor Kompas HMC4883L

Dari data pengujian Tabel 4.1, sudut kompas yang terbaca diantara  $0 < \text{sudut} < 360$ . Pengukuran kompas acuan dengan hasil pengukuran kompas HC5883L memiliki persentase *error* kecil. Tetapi ada beberapa pembacaan sudut yang menunjukkan persentase *error* besar diantaranya :  $30^\circ$ ,  $60^\circ$ ,  $90^\circ$ ,  $150^\circ$ , dan  $250^\circ$ . Hal ini disebabkan pada sudut tersebut adanya medan magnet yang dapat mengganggu pembacaan sensor kompas HC 5883L. Pada Gambar 4.3 menunjukkan pembacaan sensor kompas mendekati linear, hal tersebut dibuktikan dengan hasil pengukuran dan hasil persamaan linear dengan nilai  $x$  dan konstanta yang kurang dari 5.

#### 4.2. Pengujian Sensor GPS Ublox Neo M8

pengujian GPS dengan cara membandingkan nilai GPS acuan dengan nilai GPS yang dihasilkan oleh sensor GPS Ublox Neo M8. Data perbandingan GPS yang diakses yaitu *longitude* dan *latitude*. Pengujian mengambil beberapa titik lokasi berbeda di Gedung AJ dan Gedung B Teknik Elektro ITS.

Tabel 4.2 Pengujian sensor GPS dengan GPS acuan

No	GPS Acuan		Sensor GPS		Error		
	Latitude	Longitude	Latitude	Longitude	Latitude(%)	Longitude(%)	Selisih (Meter)
1	-7,285,088	112,796,168	-7,285,095	112,796,173	0.0000961	0.000004433	5.44
2	-7,285,001	112,796,290	-7,285,006	112,796,287	0.0000686	0.000002660	2.67
3	-7,285,110	112,796,430	-7,285,116	112,796,432	0.0000824	0.000001773	2.38
4	-7,285,055	112,796,405	-7,285,059	112,796,401	0.0000549	0.000003546	3.73
5	-7,285,030	112,796,344	-7,285,037	112,796,340	0.0000961	0.000003546	3.54
6	-7,285,108	112,796,406	-7,285,111	112,796,401	0.0000412	0.000004433	4.80
7	-7,285,125	112,796,376	-7,285,127	112,796,371	0.0000275	0.000004433	4.86
8	-7,285,230	112,795,810	-7,285,233	112,795,814	0.0000412	0.000003546	4.19
9	-7,285,097	112,795,891	-7,285,090	112,795,898	0.0000961	0.000006206	6.53
10	-7,285,174	112,796,474	-7,285,167	112,796,470	0.0000961	0.000003546	4.44

Dari data pengujian *latitude* dan *longitude*, dapat dianalisa bahwa terdapat kesalahan pada data GPS terhadap GPS acuan dari Google Map. Dari data Tabel 4.2 rata-rata selisih jarak dari pengujian ini yaitu 4,26 meter, terdapat selisih mendekati 7 meter. Hal tersebut dikarenakan faktor lingkungan, cuaca dan jenis sensor modul GPS yang digunakan. Dimana saat pengujian, lokasi pengujian agak tertutup dan hari tidak begitu cerah sehingga pembacaan sinyal GPS *satelite* terganggu. Jenis modul GPS yang digunakan yaitu modul GPS yang terintegrasi sebagai modul mikrokontroler Arduino, hanya bisa berkerja pada *baudrate* 9600 bps saja dan memiliki toleransi pembacaan *error*  $\pm 5$  meter.

Untuk mengatasi *error* yang cukup besar tersebut dapat dilakukan penggantian jenis modul sensor GPS yang digunakan seperti GPS NEO U-blox M8N APM dan PIXHAWK. Modul GPS tersebut memiliki akurasi yang cukup tinggi dengan *error* 0,6 meter dan dapat berkerja pada *baudrate* 38400 bps serta dilengkapi dengan kompas digital HMC5833L. Selain itu, pengujian dilakukan ditempat yang lebih terbuka seperti tanah lapang sehingga pembacaan sensor GPS tidak terganggu yang dapat menekan kesalahan pembacaan sensor GPS.

#### 4.3. Pengujian Sensor Gas

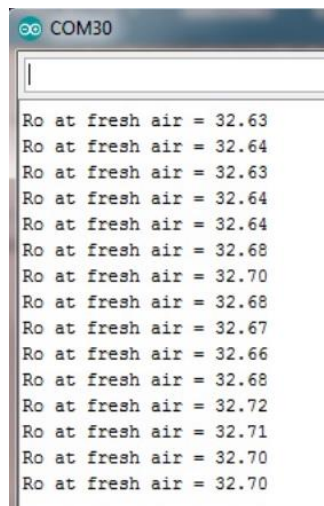
Pengujian dilakukan dengan menggunakan gas yang didapat dari gas Iso-Butana dan sudah memiliki tekanan meskipun lemah. Pengujian dilakukan di ruangan B402, suhu ruangan 27°C dan jarak gas dengan sensor 30 cm terlihat pada Gambar 4.4. Dari pengujian ini akan dapat diketahui respon sensor terhadap perubahan konsentrasi gas di udara. Perbandingan  $R_s / R_o$  pada saat udara bersih apabila dilihat dari grafik perbandingan  $R_s / R_o$  yaitu 20. Dari hasil perbandingan tersebut didapatkan nilai  $R_o$  secara konstant ketika gas mendeteksi udara bersih. Pada gambar 4.5 merupakan hasil nilai percobaan pembacaan  $R_o$  rata-rata pada saat udara bersih yaitu 32 K $\Omega$ . Data hasil pengujian respon sensor berupa perubahan perbandingan  $R_s / R_o$  dan perubahan konsentrasi gas terhadap waktu  $t$  (ms).

Dari pengujian ini dapat disimpulkan bahwa pendeteksian gas di udara memiliki respon sesuai dengan Gambar 3.10. Akan tetapi, ketika sensor sudah mendeteksi adanya perubahan konsentrasi gas di udara sekitarnya, maka untuk bisa kembali ke keadaan normal waktu yang cukup lama (pada saat  $t > 50$  detik). Pada pengujian ini, sensor mendeteksi konsentrasi gas di udara

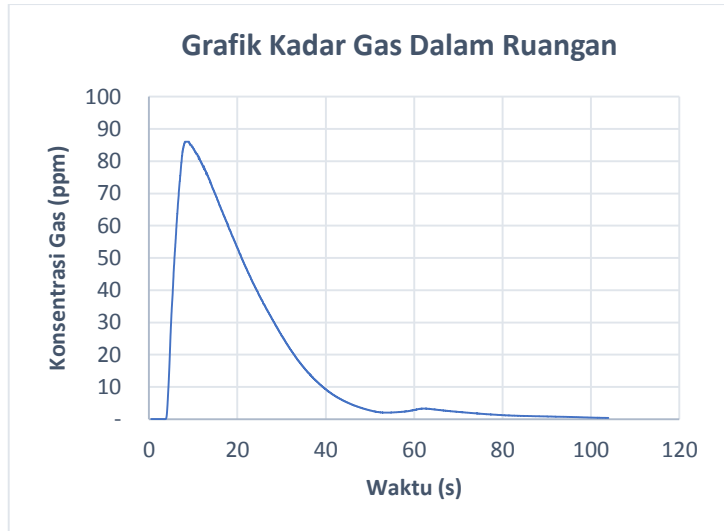
membutuhkan waktu pada saat  $1 < t < 40$  detik. sehingga untuk mengatasi hal ini maka diperlukan sirkulasi udara yang lebih baik agar konsentrasi udara yang telah tercemar oleh gas yang akan terdeteksi kembali normal dengan cepat.



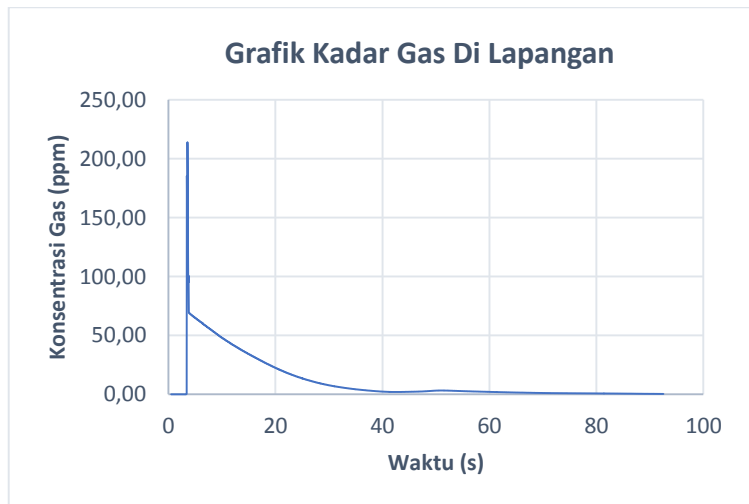
Gambar 4.4 Pengujian sensor gas dalam ruangan



Gambar 4.5 Nilai pembacaan Ro saat udara bersih



Gambar 4.6 Grafik konsentrasi gas dalam ruangan



Gambar 4.7 Grafik konsentrasi gas di lapangan

Hasil pengujian pembacaan konsentrasi gas pada lapangan dapat diperlihatkan pada Gambar 4.7, dari hasil tersebut dapat dilihat bahwa pembacaan perubahan konsentrasi gas di udara sekitar sensor membutuhkan waktu lebih cepat yaitu  $1 < t < 30$  detik. Akan tetapi nilai kadar gas konsentrasi yang dibaca rata-rata lebih rendah yaitu 60 ppm. Hal tersebut dikarenakan saat pengujian di lapangan gas yang dibaca oleh sensor terbawa oleh angin sehingga menyebabkan sensor membaca konsentrasi gas lebih rendah dibandingkan dalam ruangan. Hal tersebut dapat dilihat juga pada sensor gas kembali ke keadaan normal lebih cepat yaitu  $t > 40$  detik.

#### **4.4. Pengujian Panel Surya**

Pengujian panel surya disini dimaksudkan untuk mengetahui sejak kapan panel surya mampu memberikan suplai daya sebagai sumber energi baterai. Pengujian dilakukan di depan Gedung AJ201 Teknik Elektro FTE ITS. Panel surya diposisikan seperti posisi saat dipasang pada *Mobile Robot* yaitu dengan kemiringan sekitar 0 derajat seperti yang terlihat pada gambar 4.13. Dari hasil pengujian, diketahui bahwa panel surya dapat menghasilkan arus mulai jam 08:00 hingga 17:30 WIB. Daya yang dihasilkan bergantung pada intensitas cahaya. Dalam grafik pada gambar 4.14 diketahui bahwa arus maksimal yang dapat dihasilkan panel surya adalah sebesar 0.9 A, yakni mulai jam 12:00 hingga 13:30 WIB dengan intensitas cahaya rata-rata 9620 lux. Tegangan yang dihasilkan panel surya ini sebesar 15.30 Volt. Sehingga, potensi daya maksimum dari modul panel surya ini adalah 13.59 Watt. Adapun 5 jam didapat dari efektivitas rata-rata waktu sinar matahari bersinar yang diserap oleh panel surya yang di dapatkan dari Tabel 4.3, pengamatan efektivitas rata-rata yang telah di uji coba di lapangan secara berkala setiap 30 menit.

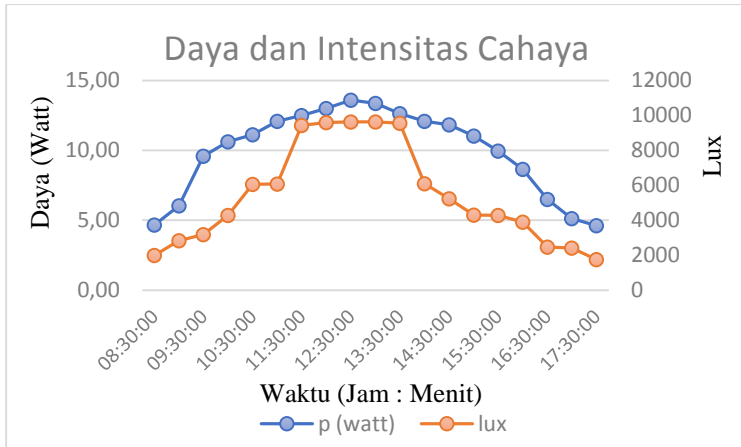




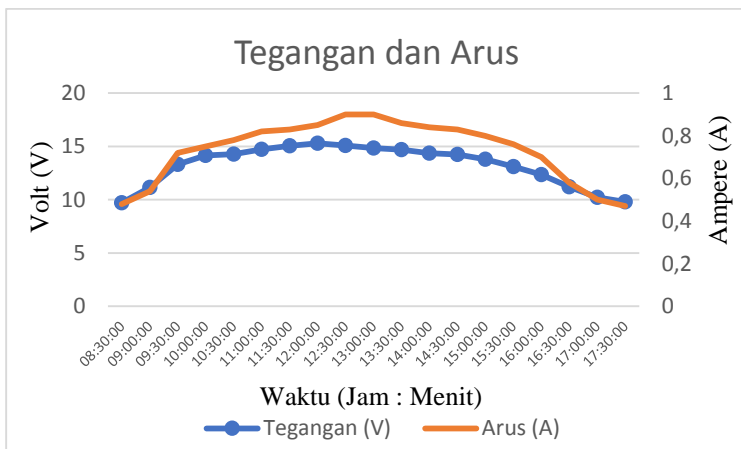
Gambar 4.8 Pengujian panel surya

Tabel 4.3 Pengukuran daya maksimum PV

Waktu	V (volt)	I (A)	P (watt)	Lux
08:30:00	9.71	0.48	4.66	1997
09:00:00	11.16	0.54	6.03	2824
09:30:00	13.30	0.72	9.58	3189
10:00:00	14.15	0.75	10.61	4273
10:30:00	14.27	0.78	11.13	6065
11:00:00	14.73	0.82	12.08	6070
11:30:00	15.06	0.83	12.50	9432
12:00:00	15.30	0.85	13.01	9589
12:30:00	15.10	0.90	13.59	9625
13:00:00	14.84	0.90	13.36	9623
13:30:00	14.68	0.86	12.62	9561
14:00:00	14.37	0.84	12.07	6102
14:30:00	14.25	0.83	11.83	5224
15:00:00	13.80	0.80	11.04	4306
15:30:00	13.10	0.76	9.96	4285
16:00:00	12.34	0.70	8.64	3891
16:30:00	11.20	0.58	6.50	2473
17:00:00	10.23	0.5	5.12	2408
17:30:00	9.80	0.47	4.61	1757



Gambar 4.9 Grafik pengukuran daya terhadap intensitas cahaya



Gambar 4.10 Grafik pengukuran tegangan terhadap arus

Dari hasil pengujian di lapangan, efisiensi daya yang dihasilkan oleh modul panel surya adalah 67.95%. Pada perancangan diketahui bahwa kebutuhan daya minimum yang digunakan *Mobile Robot* adalah sebesar 17.94 Watt. Berdasarkan hasil ini, maka penggunaan panel surya hanya membantu baterai untuk *charging* baterai ketika mengangkat beban listrik dari *Mobile Robot*. Dengan hal tersebut, konsumsi daya pada *Mobile Robot* akan lebih tahan lama.

Saat pengujian ketika baterai robot habis, maka robot akan berhenti untuk sementara kemudian menunggu baterai charging oleh panel surya. *Mobile Robot* bisa berjalan kembali ketika baterai robot sudah penuh diisi oleh panel surya.

#### 4.5. Pengujian Baterai

Setelah beberapa percobaan yang telah dilakukan pada panel surya tentang berapa maksimum arus yang dapat disuplai untuk mengisi baterai *Mobile Robot*. Pada pengujian baterai *Mobile Robot* ini dilakukan dua pengujian diantaranya : untuk mengetahui beberapa lama durasi yang dibutuhkan oleh panel surya saat mengisi arus dari baterai dan beberapa lama durasi baterai habis saat robot berkerja di lapangan dengan menggunakan bantuan panel surya dan tanpa panel surya. Sebelumnya telah dilakukan pengukuran secara langsung terhadap arus total beban dari *Mobile Robot* yang dapat disajikan pada Tabel 4.4.

Tabel 4.4 Pengukuran penggunaan arus pada robot

Operasi mode	Arus rata-rata
Peralatan elektronik nyala (Arduino, raspberry pi dan sensor )	450 mA
4 motor nyala	2800 mA
Semua beban	3250 mA

Tabel 4.5 Pengukuran durasi *charging* baterai

Jam	Lux Rata-Rata	Durasi
09:00	5430	150 menit
12:00	9566	75 menit
15:00	3805	180 menit

Tabel 4.6 Pengukuran durasi *discharging* baterai

Mode	Jam	Lux Rata-Rata	Durasi
Baterai	-	-	55 menit
Baterai + PV	09:00	5430	25 menit
	12:00	9566	125 menit
	15:00	3805	20 menit

Dari Tabel pengujian di atas, diketahui arus total rata-rata yang dibutuhkan oleh *Mobile Robot* dalam beroperasi yaitu 3.25 Ampere. Sehingga catu daya dari baterai dan panel surya harus memenuhi besarnya arus tersebut. Dari kedua kombinasi catu daya tersebut digunakan untuk mensuplai arus yang dibutuhkan oleh *Mobile Robot* agar dapat berkerja lebih tahan lama. Pengujian *charging* saat tidak beroperasi dan *discharging* baterai saat beroperasi dapat disajikan pada Tabel 4.5 dan Tabel 4.6.

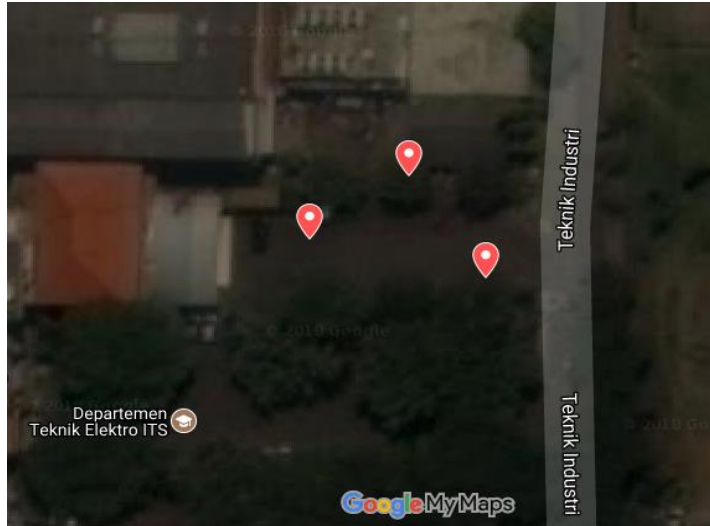
Pada hasil pengujian *discharging* baterai, dapat diketahui bahwa ketika beroperasi tanpa menggunakan panel surya, *Mobile Robot* dapat beroperasi 55 menit dari 48 menit yang direncanakan. Tetapi ketika dibantu dengan panel surya *Mobile Robot* dapat bertahan maksimum 125 menit pada jam 12:00, karena pada saat itu panel surya menghasilkan arus maksimum untuk mengisi baterai. Hal tersebut dapat dibuktikan dengan kemampuan panel surya yang dapat mengisi baterai dalam durasi 75 menit tanpa beroperasi. Sedangkan pada Jam 09.00 dan 15.00 robot hanya bertahan kurang dari 30 menit, hal tersebut diakibatkan pada jam tersebut panel surya tidak optimal dalam mensuplai daya pada baterai dan beban pada *Mobile Robot* bertambah dengan adanya panel surya dan SCC terhadap baterai sehingga membuat daya pada baterai cepat habis dibandingkan tanpa pemasangan panel surya dan SCC.

#### 4.6. Pengujian Waypoint

Pengujian navigasi *waypoint* dilakukan untuk mengetahui seberapa besar ketepatan robot dalam bernavigasi menuju titik *waypoint* yang sudah diatur akan dilalui oleh *Mobile Robot*. Titik-titik tersebut dapat disajikan pada Tabel 4.6. Titik-titik yang sudah diatur kemudian diplot pada *Google Maps* untuk menunjukan titik pada peta. Hasil plot titik tersebut dapat dilihat pada Gambar 4.9.

Tabel 4.7 Daftar *waypoint*

Waypoint	latitude	longitude
1	-7.284854	112.796867
2	-7.284968	112.796920
3	-7.284950	112.796829



Gambar 4.11 Hasil plot *Waypoint* di Google Maps

Target pencapaiannya ialah *Mobile Robot* harus melalui ketiga titik tersebut. Setelah *waypoint* dimasukan, robot dijalankan dan mengirim koordinat setiap satu detik. Hasil data pengiriman koordinat ditunjukkan pada Tabel 4.7 dan hasil plotting data disajikan pada Gambar 4.10 dan Gambar 4.11.

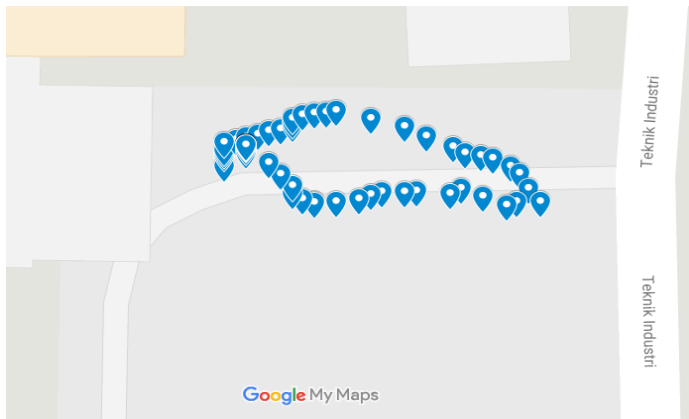
Tabel 4.8 Titik-titik koordinat *waypoint* yang dilalui robot

Waypoint	Latitude	Longitude
1	-7.284914	112.796722
2	-7.284906	112.796722
3	-7.284904	112.796722
4	-7.284902	112.796722
5	-7.284896	112.796722
6	-7.284894	112.796730
7	-7.284892	112.796737
8	-7.284890	112.796745
9	-7.284887	112.796752
10	-7.284886	112.796760
11	-7.284884	112.796768
12	-7.284881	112.796768
13	-7.284878	112.796768
14	-7.284875	112.796775
15	-7.284874	112.796783
16	-7.284873	112.796791

17	-7.284872	112.796798
18	-7.284878	112.796821
19	-7.284884	112.796844
20	-7.284891	112.796859
21	-7.284899	112.796877
22	-7.284904	112.796885
23	-7.284906	112.796896
24	-7.284908	112.796904
25	-7.284914	112.796916
26	-7.284919	112.796922
27	-7.284930	112.796928
28	-7.284940	112.796936
29	-7.284940	112.796920
30	-7.284943	112.796913
31	-7.284936	112.796897
32	-7.284930	112.796882
33	-7.284934	112.796875
34	-7.284932	112.796852
35	-7.284933	112.796844
36	-7.284933	112.796829
37	-7.284935	112.796821
38	-7.284938	112.796813
39	-7.284940	112.796798
40	-7.284940	112.796783
41	-7.284941	112.796783
42	-7.284938	112.796775
43	-7.284934	112.796768
44	-7.284928	112.796768
45	-7.284920	112.796760
46	-7.284911	112.796752
47	-7.284904	112.796737
48	-7.284901	112.796737
49	-7.284899	112.796737
50	-7.284898	112.796737
51	-7.284898	112.796737
52	-7.284898	112.796737
53	-7.284898	112.796737
54	-7.284898	112.796737
55	-7.284898	112.796737
56	-7.284898	112.796737
57	-7.284898	112.796737
58	-7.284898	112.796737
59	-7.284898	112.796737
60	-7.284898	112.796737
61	-7.284898	112.796737



Gambar 4.12 Hasil plotting koordinat yang dilalui robot *satelite view*

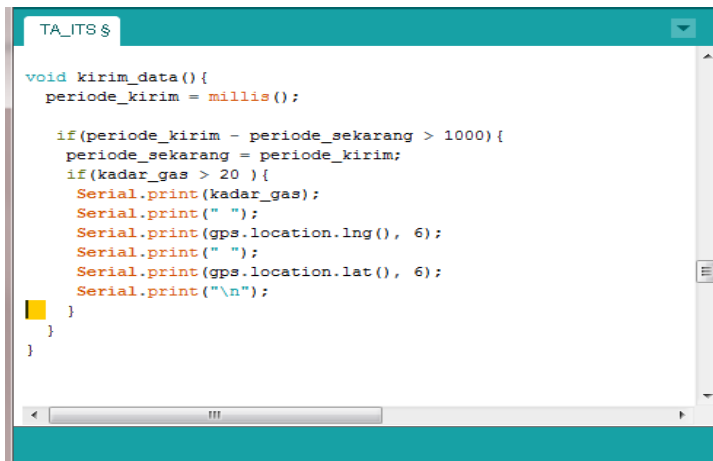


Gambar 4.13 Hasil plotting koordinat yang dilalui robot *map view*

Dari gambar di atas, terlihat bahwa titik-titik koordinat yang berhasil diplot terlihat sedikit berbeda, dalam artian ada yang terlihat melenceng. Namun, masih dalam toleransi radius 3 meter, sehingga *Mobile Robot* masih tetap bisa menuju titik-titik koordinat tersebut. Hal ini karena saat pengujian cuaca kurang bagus, yang mana dapat mempengaruhi data titik koordinat yang diterima oleh sensor GPS. Serta modul GPS yang digunakan yaitu jenis modul yang terintegrasi khusus mikrokontroler dimana memiliki toleransi *error* pembacaan  $\pm 5$  meter.

#### 4.7. Pengujian *Uploading Data ke Database*

Pengujian *upload* data ke database dimulai dari penerimaan data yang dikirim oleh mikrokontroler arduino dengan komunikasi serial oleh raspberry pi. Data yang dikirim yaitu kadar gas, *longitude* dan *latitude* GPS yang sudah diproses pada arduino. Pada Gambar 4.12 merupakan program arduino dalam mengirim data ke raspberry. Sedangkan Gambar 4.13 merupakan program python import serial dalam menerima data dari arduino oleh raspberry pi.

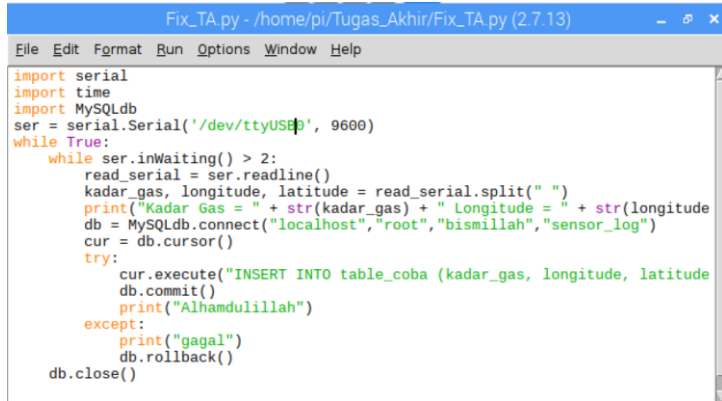
The image shows a screenshot of an Arduino IDE window. The title bar at the top reads "TA\_ITS \$". The code editor contains the following C++ code:

```
void kirim_data() {  
    periode_kirim = millis();  
  
    if(periode_kirim - periode_sekarang > 1000){  
        periode_sekarang = periode_kirim;  
        if(kadar_gas > 20 ){  
            Serial.print(kadar_gas);  
            Serial.print(" ");  
            Serial.print(gps.location.lng(), 6);  
            Serial.print(" ");  
            Serial.print(gps.location.lat(), 6);  
            Serial.print("\n");  
        }  
    }  
}
```

The code is color-coded: keywords in blue, variables and constants in black, and string literals in red. The IDE interface includes a file explorer on the right and a scroll bar at the bottom.

Gambar 4.14 Program pengiriman data GPS dan Kadar gas di Arduino

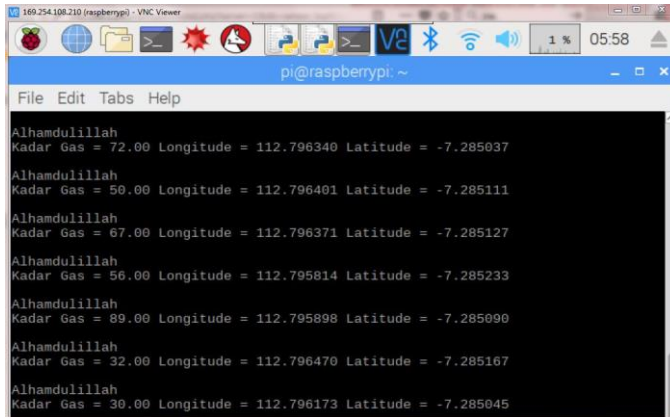


The image shows a screenshot of a text editor window with a blue title bar. The title bar text is 'Fix\_TA.py - /home/pi/Tugas\_Akhir/Fix\_TA.py (2.7.13)'. Below the title bar is a menu bar with 'File', 'Edit', 'Format', 'Run', 'Options', 'Window', and 'Help'. The main area of the window contains Python code. The code imports 'serial' and 'time' modules, and 'MySQLdb' module. It creates a serial object 'ser' with '/dev/ttyUSB0' and 9600. A 'while True' loop starts with 'ser.inWaiting() > 2:'. Inside the loop, it reads a line from the serial port, splits it into 'kadar\_gas', 'longitude', and 'latitude', and prints them. Then it connects to a MySQL database 'bismillah' on 'localhost' with user 'root', creates a cursor, and tries to insert the data into a table 'table\_coba'. If successful, it prints 'Alhamdulillah'. If an exception occurs, it prints 'gagal' and rolls back the transaction. Finally, it closes the database connection.

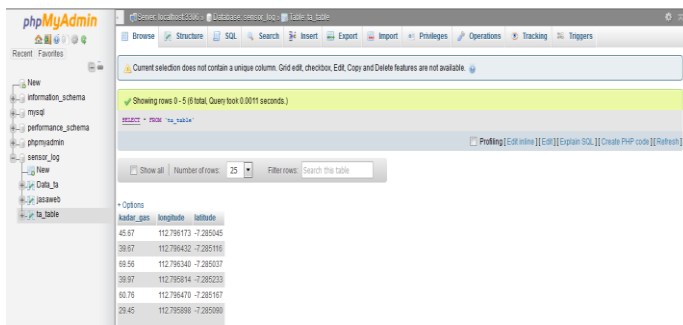
```
import serial
import time
import MySQLdb
ser = serial.Serial('/dev/ttyUSB0', 9600)
while True:
    while ser.inWaiting() > 2:
        read_serial = ser.readline()
        kadar_gas, longitude, latitude = read_serial.split(" ")
        print("Kadar Gas = " + str(kadar_gas) + " Longitude = " + str(longitude)
        db = MySQLdb.connect("localhost", "root", "bismillah", "sensor_log")
        cur = db.cursor()
        try:
            cur.execute("INSERT INTO table_coba (kadar_gas, longitude, latitude
            db.commit()
            print("Alhamdulillah")
        except:
            print("gagal")
            db.rollback()
    db.close()
```

Gambar 4.15 Program penerimaan data GPS dan Kadar gas di Python

Dari Gambar terlihat bahwa serial komunikasi antara arduino dengan raspberry pi menggunakan koneksi USB0 dan *baudrate* 9600. Apabila koneksi telah terhubung, data parameter yang diterima akan disimpan pada database server MySQL phpMyAdmin yang telah dibuat kemudian ditampilkan dalam bentuk website seperti pada Gambar 4.17. Pada saat pengujian sistem penerimaan data dan penyimpanan data pada database MySQL berjalan secara lancar, namun tergantung pada koneksi internet yang terhubung pada raspberry. Semakin bagus koneksinya maka data yang di-*upload* tidak akan mengalami keterlambatan begitu juga sebaliknya. Proses keberhasilan penyimpanan database MySQL dapat dilihat pada Serial LxTerminal pada Gambar 4.14.



Gambar 4.16 Serial LX Terminal Raspberry



Gambar 4.17 Penyimpanan database MySQL

Pembuatan database phpMyAdmin disesuaikan dengan jumlah data sensor yang diterima oleh raspberry pi. Dari hal tersebut dibuat tiga kolom tabel database yaitu kadar\_gas, longitude dan latitude. Semua kolom dibuat dengan tipe variabel *Text*.

#### 4.8. Pengujian Sistem Secara Keseluruhan

Pengujian ini adalah pengujian yang dilakukan pada sistem yang terintegrasi untuk diuji pada lapangan yang sebenarnya. Pengujian dilakukan di Departemen Teknik Elektro FTE Institut Teknologi Sepuluh Nopember. Dengan memberikan tiga titik *waypoint* yang sudah diatur yaitu : *waypoint* 1 pada lokasi - 7.284854, 112.796867; *waypoint* 2 pada lokasi - 7.284968,

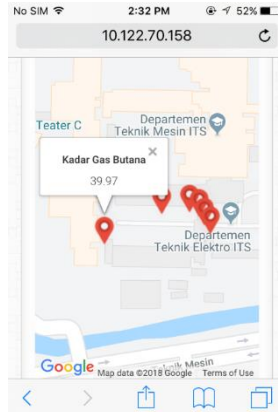
112.796920; dan *waypoint* 3 pada lokasi -7.284950, 112.796829. Selama robot melewati titik-titik *waypoint* tersebut, pada tengah-tengah perjalanan robot disemprotkan gas butana dengan manual. Konsentrasi Gas yang terdeteksi oleh sensor *mobile* robot melebihi kadar konsentrasi gas 20 ppm akan di-*upload* pada website yang sudah dibuat untuk dipetakan lokasinya. Hasil pemetaan konsentrasi gas oleh *Mobile* Robot dapat dimuat dalam Gambar 4.17 dan Tabel 4.9.

Tabel 4.9 Pemetaan konsentrasi Gas

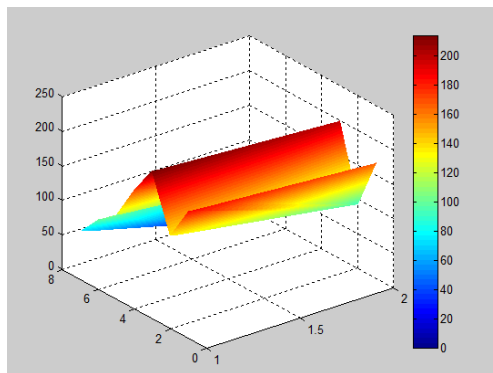
Kadar Gas	Longitude	Latitude
45.67	112.796847	-7.284824
39.67	112.796827	-7.284804
69.56	112.796890	-7.284836
39.97	112.796867	-7.284854
60.76	112.79673	-7.284923
29.45	112.795898	-7.285090



Gambar 4.18 Pemetaan Kadar gas pada web tampilan PC



Gambar 4.19 Pemetaan Kadar gas pada web tampilan *Smartphone*



Gambar 4.20 Grafik distribusi hasil pemetaan konsentasi gas di lapangan

Dari hasil pemetaan konsentrasi gas oleh *Mobile Robot*, dapat secara langsung dilihat pada web yang dapat diakses oleh *client* dengan menggunakan *device* seperti: PC (*personal Computer*) dan *Smartphone* yang terhubung dengan internet. Informasi pemetaan konsentrasi gas selalu *up to date* secara otomatis. Hasil distribusi pemetaan konsentrasi gas di lapangan menunjukkan apabila terdeteksi gas yang sangat banyak akan diidentifikasi berwarna merah, sedangkan apabila ditempat tersebut tidak ada konsentrasi gas sama sekali akan diidentifikasi berwarna biru.

## **BAB V**

### **KESIMPULAN DAN SARAN**

#### **5.1. Kesimpulan**

Dari hasil analisa pengujian *Mobile Robot Bertenaga Surya Untuk Pemetaan Konsentrasi Gas* yang terdiri dari modul panel surya, Sensor Gas TGS 2600, Sensor GPS Ublox Neo-M8N, Sensor *Compass* HMC5883L dengan mikrokontroller Arduino Mega2560 dan Raspberry Pi, database MySQL phpMyAdmin, jaringan komunikasi *wireless* dan web hosting yang terintegrasi dengan Google Maps. Didapatkan beberapa kesimpulan antara lain :

Realisasi sistem *Mobile Robot* dapat berkerja sesuai yang diharapkan yaitu dapat memetakan konsentrasi gas lebih dari 20 ppm pada tiga titik *waypoint* yang telah ditentukan dengan menampilkannya dalam bentuk web hosting secara *online realtime*. Dengan catu daya tambahan panel surya, sistem mampu berkerja lebih tahan lama 60 menit dibandingkan dengan menggunakan baterai yang berkerja sendiri pada waktu operasi jam 12:00 dengan intensitas cahaya rata-rata 9600 lux. Daya maksimum yang dapat dihasilkan oleh panel surya sebesar 13.4 Watt pada jam 11:30 – 13:30 dengan intensitas cahaya rata-rata 9600 lux. Dari pengujian *waypoint* didapatkan hasil *error* rata-rata pembacaan GPS sebesar 3 meter. Sensor Gas mampu mendeteksi kadar konsentrasi gas pada selang respon waktu 40 detik dengan jarak 30 cm dalam keadaan udara bersih di dalam ruangan sedangkan pada lapangan dalam selang waktu 30 detik.

#### **5.2. Saran**

Perlunya penggunaan SCC PWM berjenis *boost converter* atau MPPT untuk mengisi arus baterai sehingga *Mobile Robot* dapat berkerja lebih tahan lama diwaktu yang menghasilkan intensitas cahaya kurang dari 9600 lux yaitu 09:00 pagi dan 15:30 sore. Penggantian roda *Mobile Robot* dari 4 WD dengan roda robot Tank agar dapat bermanuver di segala rintangan. Perlunya grafik distribusi pemetaan konsentrasi gas pada *website* agar diketahui kadar bahaya konsentrasi gas dari hasil lokasi pemetaan tersebut.

.....*Halaman ini sengaja dikosongkan*.....

## DAFTAR PUSTAKA

- [1] H. Avrilyantama, M. Rivai, dan D. Purwanto, “Pengembangan Robot Hexapod untuk Melacak Sumber Gas,” Des 2015.
- [2] Roby Adi Wibowo, Muhammad Rivai, dan Suwito, “Impelementasi Autonomous Navigation Robot Menggunakan Global Positioning System (GPS) Untuk Pemetaan Kadar Gas Berbahaya,” Jan 2017.
- [3] M. Rossi dan D. Brunelli, “Autonomous Gas Detection and Mapping With Unmanned Aerial Vehicles,” *IEEE Trans. Instrum. Meas.*, vol. 65, no. 4, hlm. 765–775, Apr 2016.
- [4] I. Priyanta, M. Rivai, dan R. Dikairono, “Pemetaan Distribusi Gas Polutan Menggunakan Quadcopter Berbasis Autonomous Waypoint Navigation,” vol. 5, Jan 2016.
- [5] L. Aufar Akbar, M. Rivai, dan F. Budiman, “Rancang Bangun Sensor Node pada Wireless Sensor Network Menggunakan Deret Sensor Gas dan Jaringan Syaraf Tiruan untuk Mendeteksi Kebakaran Hutan,” vol. 5, Jan 2016.
- [6] M. Rivai, R. Dikairono, dan A. Tomi, “Sistem Monitoring PH dan Suhu Air dengan Transmisi Data Nirkabel,” Okt 2010.
- [7] B. Aji Supeno, M. Rivai, dan F. Budiman, “Rancang Bangun Data Logging Berbasis Web Server Pada Robot Balon Udara Untuk Deteksi Kebocoran Pipa Gas,” Jun 2016.
- [8] D. Arief Rahman K., M. Rivai, dan R. Dikairono, “Sistem Navigasi Pada Balon Udara Menggunakan GPS dan Kontrol Logika Fuzzy,” Jun 2016.
- [9] P. A. Plonski, J. V. Hook, dan V. Isler, “Environment and Solar Map Construction for Solar-Powered Mobile Systems,” *IEEE Trans. Robot.*, vol. 32, no. 1, hlm. 70–82, Feb 2016.
- [10] A. S. Prasetya, “Pengaturan Tracking Daya Maksimum dalam Sistem Pembangkitan Listrik Tenaga Surya dengan Fuzzy Takagi Sugeno,” Undergraduate, Institut Teknplogi Sepuluh Nopember, 2017.
- [11] Luthfan Aufar Akbar, Muhammad Rivai, dan Fajar Budiman, “Rancang Bangun Sensor Node pada Wireless Sensor Network Menggunakan Deret Sensor Gas dan Jaringan Syaraf Tiruan untuk Mendeteksi Kebakaran Hutan.”
- [12] Lilik Kunarso, “Rancang Bangun Sistem Kontrol Listrik Berbasis Web Menggunakan *Server Online Mini Pc Raspberry pi*” (TugasAkhir). Universitas Stikubank. Semarang. 2015.

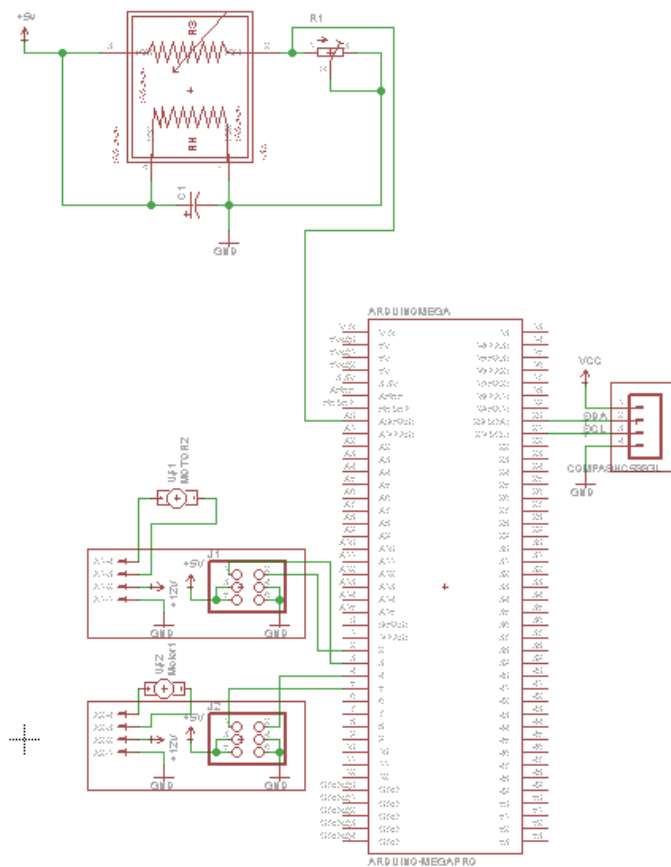
- [13] *Datasheet Rapsberry pi* <https://www.raspberrypi.org/>
- [14] Muhammad Rivai. 2016. "Sistem keamanan pada Lingkungan Pondok Pesantren menggunakan *Raspberry pi*". Institut Teknologi Sepuluh November. ISSN: 2337-3539
- [15] D. Reyes, G. Millan, R. Osorio-Corparan, dan G. Lefranc, "Mobile Robot Navigation Assisted by GPS," *IEEE Lat. Am. Trans.*, vol. 13, no. 6, hlm. 1915–1920, Jun 2015.
- [16] S. Rady, A. A. Kandil, dan E. Badreddin, "A hybrid localization approach for UAV in GPS denied areas," dalam *2011 IEEE/SICE International Symposium on System Integration (SII)*, 2011, hlm. 1269–1274.
- [17] J. Wang, P. Huang, C. Chen, W. Gu, dan J. Chu, "Stereovision aided navigation of an Autonomous Surface Vehicle," dalam *2011 3rd International Conference on Advanced Computer Control*, 2011, hlm. 130–133.
- [18] W. Lee dan W. Chung, "Position estimation using multiple low-cost GPS receivers for outdoor mobile robots," dalam *2015 12th International Conference on Ubiquitous Robots and Ambient Intelligence (URAI)*, 2015, hlm. 460–461.
- [19] W. Habibi, "Pembangunan Sistem Pelacakan dan Penelusuran Device Mobile Berbasis Global Positioning Sistem (GPS) Pada Platform Mobile Google Android," ITS, Surabaya, 2011.
- [20] U-Blox, "NEO-M8: U-Blox M8 Concurrent GNSS Modules," u-blox, South America.
- [21] F. T. Putri, M. Ariyanto, Ismoyo, M. Arozi, W. Caesarendra and M. R. Ibrahim, "*Development of Unmanned Aerial Vehicle (UAV) Ornithopter with Wireless Radio Control*," Semarang, 2016.
- [22] "72 MHz radios," The Balsa Workbench, 2017. [Online]. Available: [http://www.balsaworkbench.com/?page\\_id=337](http://www.balsaworkbench.com/?page_id=337). [Accessed 23 04 2017].
- [23] Pololu, "VNH2SP30 Motor Driver Carrier MD01B," Pololu Corporation, 2001. [Online]. Available: <https://www.pololu.com/product/706>.
- [24] UnmannedTech, "100mW Ardupilot Unmanned Telemetry Kit V2 (433MHz)," 2017. [Online]. Available: <https://www.unmannedtechshop.co.uk/100mw-ardupilot-unmanned-telemetry-kit-v2-433mhz/>. [Accessed 10 12 2017].
- [25] Anto, E. Hamdani and R. Abdullah, "Portable Battery Charger Berbasis Sel Surya," *Jurnal Rekayasa Elektrika*, vol. XI, no. 1, pp. 19-24, 2014.



- [26] Google image, "SCC PWM Buck converter", 2018. [online]. Available: [https://www.google.com/search?q=scss+pwm&client=firefox-b&source=lnms&tbm=isch&sa=X&ved=0ahUKEwj9z9i4bbbAhVQWX0KHXYAaYQ\\_AUICigB&biw=1366&bih=631#imgc=bJIo\\_2G4sxSXwM](https://www.google.com/search?q=scss+pwm&client=firefox-b&source=lnms&tbm=isch&sa=X&ved=0ahUKEwj9z9i4bbbAhVQWX0KHXYAaYQ_AUICigB&biw=1366&bih=631#imgc=bJIo_2G4sxSXwM). [Accessed 3 6 2018]
- [27] Google Image, "LIPO 2600 mAh 4s for power", 2018. [online]. Available: [https://www.google.com/search?client=firefox-b&biw=1366&bih=631&tbm=isch&sa=1&ei=kpcTW8TjNpO9QOLi6KAAg&q=lipo++power+4s+2600+mah+4power&oq=lipo++power+4s+2600+mah+4power&gs\\_l=img.3...11772.11772..0.12407.1.1.0.0.0.0.62.62.1.1.0....0...1c.1.64.img..0.0.0....0.MFL5feego9U#imgc=s1dcboCGzyehmM](https://www.google.com/search?client=firefox-b&biw=1366&bih=631&tbm=isch&sa=1&ei=kpcTW8TjNpO9QOLi6KAAg&q=lipo++power+4s+2600+mah+4power&oq=lipo++power+4s+2600+mah+4power&gs_l=img.3...11772.11772..0.12407.1.1.0.0.0.0.62.62.1.1.0....0...1c.1.64.img..0.0.0....0.MFL5feego9U#imgc=s1dcboCGzyehmM). [Accessed 3 6 2018].
- [28] M. H. Assiddiqi, "Rancang Bangun Robot Pembersih Lantai Kolam Renang," Departemen Teknologi Elektro, ITS, Surabaya, 2017.
- [29] Hartanto, "Sukses Tembus Perguruan Tinggi Negeri" Kaifa Learning, Bandung, 2011
- [30] Chapman, Stephen .J, "Electric Machinery Fundamentals 4th edition", The McGraw-Hill, New York, 2005. ISBN : 0072465239

.....*Halaman ini sengaja dikosongkan*.....

## LAMPIRAN A SKEMATIK RANGKAIAN



.....*Halaman ini sengaja dikosongkan*.....

## LAMPIRAN B

### PROGRAM ARDUINO

```
#include "I2Cdev.h"
#include "HMC5883L.h"
#include <Wire.h>
#include <SPI.h>
#include <TinyGPS++.h>
#include <math.h>
#include <string.h>
#include <stdlib.h>

unsigned long periode_kirim;
unsigned long periode_sekarang = 0;

static const uint32_t GPSBaud = 9600;
TinyGPSPlus gps;
HMC5883L mag;

int16_t mx, my, mz;
float derajat_kompas;
float awal_ADC, ro, kadar_gas;
float jarak_titik, beda_sudut;
double derajat_titik;
float PID;
int  pwm_kanan, pwm_kiri;
int point = 1;
void setup() {
    // put your setup code here, to run once:
    Wire.begin();
    pinMode(A0, INPUT);
    pinMode(2, OUTPUT);
    pinMode(3, OUTPUT);
    pinMode(4, OUTPUT);
    pinMode(5, OUTPUT);
    pinMode(6, OUTPUT);
    pinMode(7, OUTPUT);
    awal_ADC = analogRead(A0);
    float tegangan = awal_ADC * (5.0 / 1023.0);
    ro = ((4.9 * 10000.0) / tegangan) - 10000.0;
```

```

Serial.begin(9600);
Serial1.begin(GPSBaud);

mag.initialize();
// Serial.println(mag.testConnection() ? "HMC5883L connection
successful" : "HMC5883L connection failed");
// Serial.print(awal_ADC);
digitalWrite(2,HIGH);
digitalWrite(3,HIGH);
delay(500);

}
static void smartDelay(unsigned long ms) {
    unsigned long start = millis();
    do {
        while (Serial1.available())
            gps.encode(Serial1.read());
    } while (millis() - start < ms);
}
void loop() {
    float kp, ki, kd, dt;
    int last_error;

    kp = 5.0;
    ki = 0.000035;
    kd = 7.2;

    int lastError;
    unsigned long waktu_sekarang;
    unsigned long waktu_lalu;

    double titik_lat;
    double titik_lon;

    switch(point){
        case 1: titik_lat = -7.284854;
            titik_lon = 112.796867;
            break;
        case 2: titik_lat = -7.284968;
            titik_lon = 112.796920;
            break;
    }
}

```

```

    case 3: titik_lat = -7.284923;
        titik_lon = 112.796730;
        break;
    }

    pembacaan_kadarGas();
    kompas();

    jarak_titik = (unsigned
long)TinyGPSPlus::distanceBetween(gps.location.lat(),
gps.location.lng(),titik_lat, titik_lon);
    derajat_titik = TinyGPSPlus::courseTo(gps.location.lat(),
gps.location.lng(), titik_lat, titik_lon);
    smartDelay(0);

    beda_sudut = derajat_kompas - derajat_titik;

    if(beda_sudut < -180){
        beda_sudut += 360;
    }
    else if(beda_sudut > 180){
        beda_sudut -= 360;
    }

    waktu_sekarang = millis();
    dt = (float)(waktu_sekarang - waktu_lalu);

    float P = beda_sudut * kp;
    float I = ((beda_sudut*dt) + I) * ki;
    float D = ((beda_sudut - last_error) * kd) / dt;

    PID = P + I + D;

    last_error = beda_sudut;

    waktu_lalu = waktu_sekarang;

    pwm_kanan = (255 + PID);
    pwm_kiri = (255 - PID);

    if(pwm_kanan > 255){

```

```

    pwm_kanan = 255;

    if(pwm_kiri < 0){
        belok_kiri();
    }
    else{
        maju();
    }
}

if(pwm_kiri > 255){
    pwm_kiri = 255;
    if(pwm_kanan < 0){
        belok_kanan();
    }
    else{
        maju();
    }
}

if(jarak_titik <= 7){
    point++;
    // Stop();
}

if(point > 4){
    point = 1;
}

    kirim_data();
    // tampil_display();

}

void kirim_data(){
    periode_kirim = millis();

    if(periode_kirim - periode_sekarang > 1000){
        periode_sekarang = periode_kirim;
        // if(kadar_gas > 20 ){
        //   Serial.print(kadar_gas);

```



```

    // Serial.print(" ");
    Serial.print(gps.location.lng(), 6);
    Serial.print(" ");
    Serial.print(gps.location.lat(), 6);
    Serial.print("\n");
  // }
}
}

void pembacaan_kadarGas() {

  float pembacaan = analogRead(A0);
  float const eksponen = 2.718282;
  float batas = awal_ADC;
  float level = map(pembacaan, batas, 1023, 0, 100);

  float tegangan_adc = pembacaan * (5.0 / 1023.0);
  float rs = (4.9 * 10000.0) / tegangan_adc;
  rs -= 10000.0;

  float perbandingan = rs / ro;
  float nilai = (-8.868) * perbandingan;
  float nilai_eksponen = pow(eksponen, nilai);
  kadar_gas = 366.84 * nilai_eksponen;

  // Serial.print(pembacaan);
  // Serial.print("\t");
  // Serial.print(perbandingan);
  // Serial.print("\t");
  //

  if(kadar_gas < 0 ){
    kadar_gas = 0;
  }
}

void kompas(){
  mag.getHeading(&mx, &my, &mz);
  float heading = atan2(my, mx);
  if(heading < 0){
    heading += 2 * M_PI;}
}

```

```

    derajat_kompas = heading * (180/M_PI);
}

```

```

void tampil_display(){
    if (gps.location.isValid())
    {
        Serial.print(gps.location.lat(), 6);
        Serial.print(F(", "));
        Serial.print(gps.location.lng(), 6);
        Serial.print("\t");
        Serial.print(jarak_titik);
        Serial.print("\t");
        Serial.print(derajat_titik);
    }
    Serial.print("\t");
    Serial.print(derajat_kompas);

    Serial.print("\t");
    Serial.println(beda_sudut);

    // Serial.print(kadar_gas);
    // Serial.print("\t");
    // Serial.println(ro);

}

```

```

void maju(){
    analogWrite(4, pwm_kanan);
    analogWrite(5, 0);

    analogWrite(6, pwm_kiri);
    analogWrite(7, 0);
}

```

```

void belok_kanan(){
    analogWrite(4, 0);
    analogWrite(5, abs(pwm_kanan-128));

    analogWrite(6, abs(pwm_kiri-128));
    analogWrite(7, 0);
}

```

```

}

void belok_kiri(){
  analogWrite(4, abs(pwm_kanan));
  analogWrite(5, 0);

  analogWrite(6, 0);
  analogWrite(7, abs(pwm_kiri));
}

void Stop(){
  analogWrite(4, 0);
  analogWrite(5, 0);

  analogWrite(6, 0);
  analogWrite(7, 0);
}

```

.....*Halaman ini sengaja dikosongkan*.....

## **LAMPIRAN C**

### **PROGRAM RASPBERRY PI (PYTHON)**

```
import serial
import MySQLdb
ser = serial.Serial('/dev/ttyUSB0', 9600)
while True:
    while ser.inWaiting() > 2:
        read_serial = ser.readline()
        kadar_gas, longitude, latitude = read_serial.split(" ")
        print("Kadar Gas = " + str(kadar_gas) + " Longitude = " +
str(longitude) + " Latitude = " + str(latitude))
        db = MySQLdb.connect("localhost", "root", "bismillah", "sensor_log")
        cur = db.cursor()
        try:
            cur.execute("INSERT INTO table_coba (kadar_gas,
longitude, latitude) VALUES ('%s','%s','%s')" %(kadar_gas,
longitude, latitude))
            db.commit()
            print("Alhamdulillah")
        except:
            print("gagal")
            db.rollback()
    db.close()
```

.....*Halaman ini sengaja dikosongkan*.....

## LAMPIRAN D PROGRAM WEBSITE

### 1. Index.php

```
<?php
$title = "Mobile Robot Bertenaga Surya Untuk Pemetaan Kadar Gas";
include_once "header.php"; ?>
<div class="row">
  <div class="col-md-12">
    <div class="panel panel-info panel-dashboard">
      <div class="panel-heading centered">
        <h2 class="panel-title"><strong> - Welcome Message -
</strong></h2>
      </div>
      <div class="panel-body">
        <div class="centered">
          <h4>Selamat Datang di Halaman Hasil Pemetaan
konsentrasi Gas</h4>
          <h4>Silahkan memilih menu diatas untuk
melanjutkan.</h4>
        </div>
      </div>
    </div>
  </div>
</div>
</div>
</div>
</div>
</div>
</div>
<?php include_once "footer.php"; ?>
```

### 2. Header.php

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="utf-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-
scale=1">
```

```

<title>Pemetaan Konsentrasi Gas</title>
<link href="css/bootstrap.min.css" rel="stylesheet">
<link rel="stylesheet" href="css/font-awesome.min.css">
<link href="css/style.css" rel="stylesheet">
<link href="css/datatables-bootstrap.css" rel="stylesheet">
<script
src="//ajax.googleapis.com/ajax/libs/jquery/2.1.1/jquery.min.js"></s
cript>
</head>
<body>
<div class="container">
<div class="row">
<div class="tengah">
<div class="head-depan tengah">
<div class="row">
<div class="col-md-1">

</div>
<div class="col-md-11">
<h4 class="judul-head">Mobile Robot Bertenaga Surya
Untuk Pemetaan Konsentasi Gas</h4>
<p><i class="fa fa-map-marker fa-fw"></i> Sistem
Informasi Yang Memuat Data Hasil Pemetaan Konsentrasi Gas</p>
</div>
</div>
<hr class="hr1 margin-b-10" />
</div>
</div>
</div>
</div>
<div class="container margin-b70">
<nav class="navbar navbar-default navbar-utama"
role="navigation">
<!-- Brand and toggle get grouped for better mobile display -->
<div class="navbar-header">
<button type="button" class="navbar-toggle" data-
toggle="collapse" data-target=".navbar-ex1-collapse">
<span class="sr-only">Status</span>
<span class="icon-bar"></span>
<span class="icon-bar"></span>
<span class="icon-bar"></span>

```



```

        </button>
    </div>

    <!-- Collect the nav links, forms, and other content for toggling -
->
    <div class="collapse navbar-collapse navbar-ex1-collapse">
        <ul class="nav navbar-nav">
            <li><a href="index.php"><i class="fa fa-home"></i>
HALAMAN DEPAN</a></li>
            <li><a href="peta.php"><i class="fa fa-map-marker"></i>
PEMETAAN KONSENTRASI GAS</a></li>
            <li><a href="about.php" data-toggle="modal" data-
target="#about"><i class="fa fa-user"></i> ABOUT</a></li>
        </ul>
    </div><!-- /.navbar-collapse -->
</nav>

```

### 3. Footer.php

```

<div class="footer footer1">
    <div class="container">
        <div class="row">
            <div class="col-md-4 col-md-offset-4">
                
                <h4 class="white">Mobile Robot Bertenaga Surya
Untuk Pemetaan Konsentrasi Gas</h4>
                <h5 class="white">Copyright &copy; Yusril Izza
(NRP. 07111645000015) 2018</h5>
            </div>
        </div>
    </div>
</div>

<!-- Modal -->
<div class="modal fade" id="about" tabindex="-1"
role="dialog" aria-labelledby="myModalLabel" aria-
hidden="true">
    <div class="modal-dialog">
        <div class="modal-content">
            <div class="modal-header">

```

```

        <button type="button" class="close" data-
dismiss="modal"><span aria-
hidden="true">&times;</span><span class="sr-
only">Close</span></button>
        <h4 class="modal-title"
id="myModalLabel">About</h4>
    </div>
    <div class="modal-body">
        <h4>MOBILE ROBOT BERTENAGA SURYA
UNTUK PEMETAAN KONSTRASI GAS</h4>
        <p>Website ini dibuat oleh Yusril Izza
(NRP.07111645000015) untuk Tugas Akhir ITS 2018.</p>
        <p>Semoga Bermanfaat :)</p>
    </div>
</div>
</div>
</div>

<script src="js/bootstrap.min.js"></script>
<script src="js/bootstrap-hover-dropdown.js"></script>
<script src="js/script.js"></script>
<script src="js/jquery.dataTables.min.js"></script>
<script src="js/datatables-bootstrap.js"></script>

</body>
</html>

```

#### 4. koneksi.php

```

<?php
$host = "localhost";
$user = "root";
$pass = "bismillah";
$name = "sensor_log";

$koneksi = mysqli_connect($host, $user, $pass, $name) or
die("Koneksi ke database gagal!");
mysqli_select_db($koneksi, $name) or die("Tidak ada
database yang dipilih!");
?>

```

## 5. **Ambildata.php**

```
<?php
include "koneksi.php";
$Q = mysqli_query($koneksi, "SELECT * FROM ta_table")or
die(mysqli_error($koneksi));
if($Q){
    $posts = array();
    if(mysqli_num_rows($Q))
    {
        while($post = mysqli_fetch_assoc($Q)){
            $posts[] = $post;
        }
    }
    $data = json_encode(array('results'=>$posts));
    echo $data;
}

?>
```

## 6. **Peta.php**

```
<?php
$title = "Peta Hasil Konsentasi Gas ";
include_once "header.php";
?>

<meta http-equiv=refresh content=10;url=peta.php>
<div class="row">
    <div class="col-md-12">
        <div class="panel panel-info panel-dash-board
centered">
            <div class="panel-heading">
                <h2 class="panel-title"><strong> - TAMPILAN
PETA - </strong></h2>
            </div>
            <div class="panel-body">
                <div id="map"
style="width:100%;height:380px;"></div>

            </div>
        </div>
    </div>
</div>

<script
src="https://maps.googleapis.com/maps/api/js?libraries=place
```

```
s&key=AIzaSyCqKQg0guRqgMHjrpWydkshY9iJY6uzVM
&callback=initMap"></script>
```

```
<script type="text/javascript">
```

```
function initialize() {
```

```
    var mapOptions = {
```

```
        zoom: 18,
```

```
        center: new google.maps.LatLng(-
7.284854,112.796867),
```

```
        disableDefaultUI: true
```

```
    };
```

```
    var mapElement = document.getElementById('map');
```

```
    var map = new google.maps.Map(mapElement,
mapOptions);
```

```
    setMarkers(map, officeLocations);
```

```
}
```

```
var officeLocations = [
```

```
<?php
```

```
$data
```

```
=
```

```
file_get_contents('http://localhost/www.mobilerobot.com/am
bildata.php');
```

```
    $no=1;
```

```
    if(json_decode($data,true)){
```

```
        $obj = json_decode($data);
```

```
        foreach($obj->results as $item){
```

```
?>
```

```
[<?php echo $item->kadar_gas ?>,<?php echo $item-
>longitude ?>,<?php echo $item->latitude ?>],
```

```
<?php
```

```
}
```

```
}
```

```
?>
```

```
];
```

```
function setMarkers(map, locations)
```

```

{
    var globalPin = 'img/penunjuk.png';

    for (var i = 0; i < locations.length; i++) {

        var office = locations[i];
        var myLatLng = new google.maps.LatLng(office[2],
office[1]);
        var infowindow = new
google.maps.InfoWindow({content: contentString});

        var contentString =
            '<div id="content">'+
            '<div id="siteNotice">'+
            '</div>'+
            '<h6 id="firstHeading" class="firstHeading">'+ ' Kadar
Gas Butana' + '</h6>'+
            '<div id="bodyContent">'+ office[0]+
            '</div>'+
            '</div>';

        var marker = new google.maps.Marker({
            position: myLatLng,
            map: map,
            title: office[0],
            icon:'img/penunjuk.png'
        });

        google.maps.event.addListener(marker, 'click',
getInfoCallback(map, contentString));
    }
}

function getInfoCallback(map, content) {
    var infowindow = new google.maps.InfoWindow({content:
content});
    return function() {
        infowindow.setContent(content);
        infowindow.open(map, this);
    };
}

```

```
initialize();
</script>
    </div>
  </div>
</div>
</div>
</div>
</div>
<?php include_once "footer.php"; ?>
```

## LAMPIRAN E DOKUMENTASI



.....*Halaman ini sengaja dikosongkan*.....



## BIODATA PENULIS



Yusril Izza, dilahirkan di LOTIM, pada tanggal 26 Juli 1992 merupakan putra bungsu dari 5 bersaudara. Penulis menamatkan sekolah di SDN 4 Danger tahun 2004. Kemudian masuk ke SMPN 1 Masbagik, tamat tahun 2008 dan melanjutkan di SMAN 1 Terara , tamat tahun 2011. Penulis diterima kuliah di Universitas Gadjah Mada jurusan D3 Teknik Elektronika pada tahun 2013 dengan beasiswa bidikmisi. Pada tahun 2016 penulis mengambil pendidikan S1 program Lintas Jalur Departemen Teknik Elektro Fakultas Teknologi Elektro Institut Teknologi Sepuluh Nopember. Selama menimba ilmu kuliah di ITS penulis mendapatkan dana beasiswa dari pemerintah NTB, selain itu penulis juga mendapatkan beasiswa PPA dua periode berturut-turut yaitu PPA 2016 dan PPA 2017. Penulis aktif mengajar, mentoring dan melakukan banyak riset dan proyek diantaranya : Mikrokontroller, Robotika, Android, IoT dan visual interface disela-sela kesibukan kuliah. Penulis juga bagian anggota dari Laboratorium Elektronika Industri dan mengambil topik Tugas Akhir di lab tersebut.

E-mail : [yusril.izza@mail.ugm.ac.id](mailto:yusril.izza@mail.ugm.ac.id)

Hp/WA : 081998973397